UNIVERSITÉ DE TUNIS

INSTITUT SUPÉRIEUR DE GESTION

THESE DE DOCTORAT

en vue de l'obtention du titre de docteur en

INFORMATIQUE DE GESTION

MODELLING INTERACTIONS BETWEEN NODES IN A CREDIBILIST SOCIAL NETWORK

BEN DHAOU SALMA

SOUTENUE LE 22 MAI 2019, DEVANT LE JURY COMPOSÉ DE:

ELOUEDI ZIED	PROFESSEUR, ISG DE TUNIS	PRÉSIDENT
BEN SAID LAMJED	PROFESSEUR, ISG DE TUNIS	RAPPORTEUR
FARAH RIADH	PROFESSEUR, ISAM DE LA MANOUBA	RAPPORTEUR
BEN AMOR NAHLA	PROFESSEUR, ISG DE TUNIS	EXAMINATEUR
BEN YAGHLANE BOUTHEINA	PROFESSEUR, IHEC DE CARTHAGE	DIRECTEUR DE THÈSE
MARTIN ARNAUD	PROFESSEUR, IUT DE LANNION	CO-DIRECTEUR DE THÈSE

Laboratoire: LARODEC

Abstract

The detection of communities in social networks has become a very important task. Indeed, as its role consists in partitioning the nodes of a network into subgroups having properties in common, this makes it possible to analyse the behaviour of the entities of the network and to predict the evolution of the latter in time.

In social networks, information about nodes, links, and messages may be imperfect. From there, the analysis of such a type of network necessitates the use of a theory of uncertainty. In this thesis, we propose three contributions applied in the framework of the theory of belief functions:

First, we were interested in showing the advantage of using evidential attributes in social networks. Indeed, we compared the results of the classification of nodes with uncertain attributes (numerical, probabilistic, evidential) generated according to the structure of the network. To do this, we considered two scenarios: attributes generated randomly and others sorted. We also performed the tests in the case of data that was noisy. In order to measure the quality of clustering results, we used normalised mutual information (NMI).

The second contribution consists on the correction of noisy information in social networks. To do this, we proposed a model based on the comparison of the calculated distances between the triplets of the network and the coherent triplets defined initially. A triplet is composed of two nodes connected to each other by a link. In order to test the proposed approach, we first tested three cases: only the nodes are noisy, only the links are noisy and finally the nodes and the links are noisy simultaneously. Then we tested the method by varying several network parameters. In order to measure the quality of the obtained results, we calculated the accuracy.

The third contribution is to detect which links are spammed in a social network. A link is considered spammed if its initial class changes according to the types of messages transiting on it. To do this, we used the theory of belief functions to combine the information of links and messages. In order to test our approach, we considered two cases: only the messages are noisy and the messages as well as the links are noisy simultaneously. The quality of the classification results was measured using accuracy, precision and recall measurements.

Keywords Social Networks Analysis, Community Detection, Spammed Link Detection, Theory of Belief Functions

Résumé

La détection de communautés dans les réseaux sociaux est devenue une tâche très importante. En effet, comme son rôle consiste à partitionner les nœuds d'un réseau en sous groupes ayant des propriétés en commun, ceci permet d'analyser le comportement des entités du réseau et de prédire l'évolution de ce dernier dans le temps.

Dans les réseaux sociaux, les informations portant sur les nœuds, liens et messages peuvent être imparfaites. A partir de là, l'analyse d'un tel type de réseaux nécessiste l'utilisation d'une théorie de l'incertain. Dans cette thèse, nous proposons trois contributions appliquées dans le cadre de la théorie des fonctions de croyance.

Tout d'abord, nous nous sommes intéressés à montrer l'avantage de l'utilisation des attributs évidentiels dans les réseaux sociaux. En effet, nous avons comparé les résultats de la classification des nœuds ayant des attributs incertains (numériques, probabilistes, évidentiels) générés en fonction de la structure du réseau. Pour ce faire, nous avons considéré deux scénarios : attributs générés aléatoirement et d'autres triés. Nous avons également effectué les tests dans le cas des données qui ont été bruitées. Afin de mesurer la qualité des résultats de la classification, nous avons utilisé l'information mutuelle normalisée.

La deuxième contribution consiste en la correction des informations bruitées dans les réseaux sociaux. Pour ce faire, nous avons proposé un modèle qui se fonde sur la comparaison des distances calculées entre les triplets du réseau et les triplets cohérents définis initialement. On appelle un triplet deux nœuds reliés entre eux par un lien. Afin de tester l'approche proposée, nous avons testé dans un premier temps trois cas : les nœuds uniquement sont bruités, les liens uniquement sont bruités et enfin les nœuds et les liens sont bruités simultanment. Ensuite, nous avons testé la méthode en faisant varier plusieurs paramètres du réseau. Dans le but de mesurer la qualité des résultats obtenus, nous avons calculé l'exactitude.

La troisième contribution consiste à détecter quels sont les liens spammés dans un réseau social. Un lien est considéré comme spammé si sa classe initiale se modifie en fonction des types de messages transitant dessus. Pour ce faire, nous avons utilisé la théorie des fonctions de croyance pour combiner les informations des liens et des messages. Dans le but de tester notre approche, nous avons considéré deux cas : les messages sont bruités uniquement et les messages ainsi que les liens sont bruités simultanément. La qualité des résultats de la classification a été mesurée en utilisant les mesures de l'exactitude, la précision et le rappel.

Mots Clés Analyse des Réseaux Sociaux, Détection de Communautés, Détection de Liens Spammés, Théorie des Fonctions de Croyance

Acknowledgments

J'ai appris que l'on mesure le succès non pas par la situation que l'on a atteinte dans la vie, mais par les obstacles que l'on a surmontés pour essayer de réussir.
Booker T. Washington

I am grateful to many people for their support and kindness during my Ph.D. study and research at team DRUID, IRISA, University of Rennes 1 in France and at the Laboratory of Operational Research of Decision and Control of Process (LARODEC), ISG of Tunis in Tunisia.

I would like to express my sincere thanks to my supervisor in the University of Tunis, Prof. Boutheina Ben Yaghlane and my supervisors in University of Rennes 1, Prof. Arnaud Martin and Dr. Mouloud Kharoune. They have provided me with a lot of invaluable guidance and discussions. Special thanks for you, for your patience, productive comments, fruitful discussions and advices.

I am sincerely grateful to my parents who have always believed in me and in my abilities. Special thanks for you, for your unconditional support and love.

Finally, I would like to thank my friends who have always been there to support me in the most difficult moments. Your friendship is a precious gift from heaven.

Contents

Introduction **1**

Thesis Outline 5

1 Social Networks **7**

1.1 Social Networks Representation and Analysis 8

1.1.1 Social Network 8

1.1.2 Social Network Analysis 9

1.2 Uncertainty in Social Network 13

1.2.1 Basics of the Theory of Belief Functions 14

1.2.2 Related Works to the Uncertainty in Social Networks . . . 22

1.3 Clustering in Social Networks 24

1.3.1 Some Community Detection Methods with only Graphs Structures 24

1.3.2 Some Community Detection Methods with Graphs Structure and Attributes 26

1.3.3 Homophilic Behaviours in Social Networks 27

1.3.4 K-Medoids algorithm 28

1.3.5 Link Prediction Problem in Social Networks 29
1.3.6 Detection of Spammers in Social Networks 30
1.3.7 Walktrap Approach 32
1.3.8 Clustering Results Evaluation Metrics 33
1.4 Used Network 36
1.4.1 Karate Club Network 36
1.4.2 Dolphins Network 36
1.4.3 Books about US Politics Network 37
1.4.4 LFR Networks 38
1.5 Conclusion 39

2 The Advantage of Evidential Attributes in Social Networks 41
2.1 Introduction 41
2.2 Clustering based on Nodes Attributes 42
2.3 Experimentations 45
2.3.1 Results Before Adding the Noise 48
2.3.2 Results After Adding the Noise 52
2.4 Clustering Results Comparison with various Metrics 61
2.4.1 Dolphins Network 62
2.4.2 LFR Network 63
2.5 Simple Support Functions Results 64
2.6 Results given by the Walktrap Approach 66
2.7 Conclusion 67

3 An Evidential Method for Correcting Noisy Information 68
3.1 Introduction 68

3.2 Noisy Information Correction based on Nodes and Links Attributes 69
3.2.1 Noise and Consistency 69
3.2.2 Formalization 70
3.2.3 Main Steps of the Algorithm 71
3.3 Experiments 76
3.3.1 Process of Experiments 76
3.3.2 Possible Corrections 78
3.3.3 Convergence 79
3.3.4 Baseline 79
3.3.5 Improvement Rate 83
3.3.6 Experiments on Real Data: Karate Club 85
3.3.7 Experiments on LFR 87
3.3.8 LFR: Variation of the Communuities Number 93
3.3.9 LFR: Variation of the Network Size 94
3.3.10 LFR: Variation of the Mixing Parameter 96
3.3.11 Comparison of the Execution Time 98
3.4 Conclusion 99

4 A Belief Approach for Detecting Spammed Links 100

4.1 Introduction 100
4.2 Spammed Links Detection based on Nodes, Links and Messages Attributes 102
4.3 Experimentations 104
4.3.1 Baseline 106
4.3.2 Case of noisy messages only 109
4.3.3 Case of noisy messages and noisy links 110

4.3.4 Detection of Spammed Links 112
4.4 Conclusion 118

Conclusion **120**

A LFR Parameters **124**

B Results Before Adding Noise **126**
B.1 LFR Network: 50 Nodes +3 Communities 126
B.2 LFR Network: 99 Nodes + 3 Communities 127
B.3 LFR Network: 200 Nodes + 3 Communities 128

References **130**

List of Figures

1.1 An Attributed Graph. 9

1.2 The Karate Club Network. 37

1.3 Dolphins Network. 38

1.4 Books about US Politics Network. 39

2.1 Graph with Evidential Attributes on Nodes. 42

2.2 Graph with Probabilistic Attributes on Nodes. 42

2.3 Noisy Karate: First Scenario. 53

2.4 Noisy Dolphins: First Scenario. 54

2.5 Noisy Books: First Scenario. 54

2.6 Noisy LFR 50N: First Scenario. 55

2.7 Noisy LFR 99N: First Scenario. 55

2.8 Noisy LFR 200N: First Scenario. 56

2.9 Noisy LFR 300N: First Scenario. 56

2.10 Noisy karate: Second Scenario. 57

2.11 Noisy Dolphins: Second Scenario. 58

2.12 Noisy Books: Second Scenario. 58

2.13 Noisy LFR 50N: Second Scenario. 59

2.14 Noisy LFR 99N: Second Scenario. 59

2.15 Noisy LFR 200N: Second Scenario. 60

2.16 Noisy LFR 300N: Second Scenario. 60

2.17 Noisy LFR 4 Communities: Second Scenario. 64

2.18 Noisy LFR 5 Communities: Second Scenario. 65

2.19 Noisy LFR 6 Communities: Second Scenario. 65

3.1 Triplet k. 70

3.2 LFR: corrected nodes and links: case of 30 noisy nodes and 50 noisy links. 80

3.3 Probabilistic Triplet. 80

3.4 Karate Club: comparison of probabilistic and evidential accuracy: case of noisy nodes. 86

3.5 Karate Club: comparison of probabilistic and evidential accuracy: case of noisy links. 87

3.6 Karate Club: comparison of probabilistic and evidential accuracy: case of noisy nodes and links. 88

3.7 LFR: comparison of probabilistic and evidential accuracy: case of noisy nodes. 90

3.8 LFR: comparison of probabilistic and evidential accuracy: case of noisy links. 91

3.9 LFR: comparison of probabilistic and evidential accuracy: case of noisy nodes and links. 92

3.10 LFR: comparison of probabilistic and evidential accuracy: case of noisy nodes and links. 94

3.11 LFR: comparison of probabilistic and evidential accuracy: case of variation of the size of the network. 96

3.12 LFR: comparison of probabilistic and evidential accuracy: case of variation of the mixing parameter. 98

4.1 An Evidential Graph. 103

4.2 Process of the belief approach 103

4.3 A Probabilistic Graph. 107

4.4 Process of the probabilistic approach 107

4.5 Spammed links after 5 iterations: case of noisy messages only. . . 110

4.6 Spammed Links after 10 iterations: case of noisy messages only. . 110

4.7 Spammed links after 15 iterations: case of noisy messages only. . 111

4.8 Spammed links after 20 iterations: case of noisy messages only. . 111

4.9 Spammed links after 5 iterations: case of noisy messages and links. 112

4.10 Spammed links after 10 iterations: case of noisy messages and links. 112

4.11 Spammed links after 15 iterations: case of noisy messages and links. 113

4.12 Spammed links after 20 iterations: case of noisy messages and links. 113

4.13 Accuracy Results: Case of $PNCUPC$. 114

4.14 Accuracy Results: Case of PNC, PC, and $PNCUPC$. 115

4.15 Accuracy Results: Case of random and $PNCUPC$ messages. . . . 116

4.16 Precision and Recall Results at first and tenth iterations. 118

4.17 Comparison of the precision/recall results at the first and the tenth iteration. 118

List of Tables

2.1 NMI Averages et Intervals of Confidence- Case of Karate Club: First Scenario. 48

2.2 NMI Averages et Intervals of Confidence- Case of Karate Club: Second Scenario. 49

2.3 NMI Averages et Intervals of Confidence- Case of Dolphins Network: First Scenario. 49

2.4 NMI Averages et Intervals of Confidence- Case of Dolphins Network: Second Scenario. 50

2.5 NMI Averages et Intervals of Confidence- Books about US Politics Network: First Scenario. 50

2.6 NMI Averages et Intervals of Confidence- Books about US Politics Network: Second Scenario. 51

2.7 NMI Averages et Intervals of Confidence- LFR 300 Nodes: First Scenario. 52

2.8 NMI Averages et Intervals of Confidence- LFR 300 Nodes: Second Scenario. 52

2.9 Community Structures Comparison using various Metrics: Case of Dolphins Network-First Scenario 62

2.10 Community Structures Comparison using various Metrics: Case of Dolphins Network-Second Scenario 63

2.11 Community Structures Comparison using various Metrics: Case of LFR 50N-First Scenario 63

2.12 Community Structures Comparison using various Metrics: Case of LFR 50N-Second Scenario 64

2.13 NMI Results given by the Walktrap Method. 66

3.1 Coherent Triplets For 3 Communities. 77

3.2 Improvement Rate: Case of Noisy Nodes Only. 83

3.3 Improvement Rate: Case of Noisy Links Only. 84

3.4 Improvement Rate for Nodes: Case of Noisy Nodes and Noisy Links. 84

3.5 Improvement Rate for Links: Case of Noisy Nodes and Noisy Links. 84

3.6 Accuracy Average and Interval of Confidence: Case of Noisy Nodes Only in the Karate Club. 85

3.7 Accuracy Average and Interval of Confidence: Case of Noisy Links Only in the Karate Club. 86

3.8 Accuracy Average and Interval of Confidence: Case of Noisy Nodes and Links in the Karate Club. 89

3.9 Accuracy Average and Interval of Confidence: Case of Noisy Nodes Only in LFR. 90

3.10 Accuracy Average and Interval of Confidence: Case of Noisy Links Only in LFR. 91

3.11 Accuracy Average and Interval of Confidence: Case of Noisy Nodes and Noisy Links in LFR. 93

3.12 Accuracy Average and Interval of Confidence: Case of Noisy Nodes and Noisy Links-Communities Variation. 95

3.13 Accuracy Average and Interval of Confidence: Case of Noisy Nodes and Noisy Links-Network Size Variation. 97

3.14 Accuracy Average and Interval of Confidence: Case of Noisy Nodes and Noisy Links-Mixing Parameter Variation. 97

3.15 Comparison of probabilistic and evidential execution time 98

4.1 Definition of function Γ given the correspondences between $\Omega_L \times \Omega_M$ and Ω_L 105

A.1 Parameters of LFR 125

B.1 NMI Averages et Intervals of Confidence- Case LFR 50 Nodes: First Scenario. 126

B.2 NMI Averages et Intervals of Confidence- Case LFR 50 Nodes: Second Scenario. 127

B.3 NMI Averages et Intervals of Confidence- LFR 99 Nodes: First Scenario. 128

B.4 NMI Averages et Intervals of Confidence- LFR 99 Nodes: Second Scenario. 128

B.5 NMI Averages et Intervals of Confidence- LFR 200 Nodes: First Scenario. 129

B.6 NMI Averages et Intervals of Confidence- LFR 200 Nodes: Second Scenario. 129

List of Algorithms

2.1 Generation of Evidential Attributes 44
2.2 Adding Noisy Attributes 45
3.1 An Evidential Approach for Correcting Noise 75
3.2 A Probabilistic Approach for Correcting Noise 83

Abbreviations and Notations

In the following, a list as exhaustive as possible of abbreviations and notations used in this thesis:

Belief Functions and Probabilities

- Ω_N: is the frame of discernment of the nodes.
- Ω_L: is the frame of discernment of the links.
- Ω_M: is the frame of discernment of the messages.
- $\Omega_L \times \Omega_M$: is the Cartesian product of Ω_L and Ω_M.
- $\Omega_N \times \Omega_L$: is the Cartesian product of Ω_N and Ω_L.
- $\uparrow$: is the vacuous extension.
- $BetP$: is the pignistic probability.
- $m_{k_1}^{\Omega_N}$: is the mass function of the node V_{k_1}.
- $m_{k_2}^{\Omega_N}$: is the mass function of the node V_{k_2}.
- $m_{k_{12}}^{\Omega_L}$: is the mass function of the link $V_{k_{12}}$.
- $m_{C_i}^{\Omega_N}$: is the categorical mass function associated to the nodes belonging to the community C_i.

- $m^{\Omega_L}_{IC_i}, m^{\Omega_L}_{BC}$: are the categorical mass functions associated respectively to the links belonging to C_i and the links connecting two communities.
- $m^{\Omega_N}_{k_{1d}}$: is the mass function of the node V_{k_1} obtained from the calculation of the distance of Jousselme.
- $m^{\Omega_N}_{k_{2d}}$: is the mass function of the node V_{k_2} obtained from the calculation of the distance of Jousselme.
- $m^{\Omega_L}_{k_{12d}}$: is the mass function of the link $L_{k_{12}}$ obtained from the calculation of the distance of Jousselme.
- $P^{\Omega_N}_{C_i}$: is a probability on a certain event of the nodes belonging to C_i.
- $P^{\Omega_L}_{IC_i}, P^{\Omega_L}_{BC}$: are the probabilities on a certain events associated respectively to the links belonging to C_i and the links connecting two communities.
- $P^{\Omega_N}_{k_1}$: is the probability of the node V_{k_1}.
- $P^{\Omega_N}_{k_2}$: is the probability of the node V_{k_2}.
- $P^{\Omega_L}_{k_{12}}$: is the probability of the link $L_{k_{12}}$.
- $P^{\Omega_N}_{k_{1d}}$: is the probability of the node V_{k_1} obtained from the calculation of the Euclidean distance.
- $P^{\Omega_N}_{k_{2d}}$: is the probability of the node V_{k_2} obtained from the calculation of the Euclidean distance.
- $P^{\Omega_L}_{k_{12d}}$: is the probability of the link $L_{k_{12}}$ obtained from the calculation of the Euclidean distance.

Social Network

- $G = \{V^b, E^b\}$: is the evidential graph. V^b denotes the set of nodes and E^b denotes the set of edges.
- k: is a triplet of the graph.
- V_{k_1}: is the first node of the triplet k.
- V_{k_2}: is the second node of the triplet k.

- $L_{k_{12}}$: is the link connecting V_{k_1} and V_{k_2}.
- C_i: is the community i with $i = 1, ..., N$
- L: is the number of links.
- N: is the number of the communities.
- M: is the number of the triplets in the network.

Distances

- d_E: is the Euclidean distance.
- d_J: is the Jousselme distance.

Other Notations

- d_k: is the average distance
- t: is an iteration

Introduction

Nowadays, the use of computer technology and Internet has become essential. Indeed, the exploitation of the internet is multiplying thanks to the messages, phone calls with or without video as well as exchanges in social networks.

This technical breakthrough allowed the development of social networks that today bring together a large community on a platform that shares everything they like or not and what they do in real time. It allows to share video, photos, texts, smiley to know the mood of the person. Social networks even include companies that want to get in touch with their target, media that share their articles, reports, and so on.

As a result, social networks became an important part of our daily lives. Therefore, it is interesting to study and analyse the types of relationships that exist in these networks in order to understand user behaviour and study the evolution of networks over time. To do so, the study of the community structure as well as the nodes and links attributes represent main characteristics that must be taken into account to analyse these networks. In fact, this will allow to infer the importance of an actor in the network (influential node) in addition of the detection of hidden and spammed links.

The social networks analysis finds its theoretical origins in the work of mathematicians on graphs (Erdos & Rényi, 1960), but the first significant developments have emerged in the social sciences (Travers & Milgram, 1967; Wasserman & Faust, 1994).

In social network analysis (Wasserman & Faust, 1994; Prell, 2012), the observed attributes of social actors are understood in terms of patterns or structures

of ties among the units. These ties may be any existing relationship between units; for example friendship, material transactions, etc.

Currently, if we observe any social network, we will soon realize that the entities composing this network are grouped, for example, according to a center of interest, a category of age, a preference, etc.

In his work, Santo Fortunato (Fortunato, 2010) explained that communities, also called clusters or modules, represent groups of vertices which probably share common properties and/or play similar roles within the graph. He argues also that the word community itself refers to a social context. In fact, people naturally tend to form groups, within their work environment, family or friends.

The role of community detection task is to highlight these groups that have formed implicitly and have shared same interests. For example, it allows:

- to identify a group of friends in a social network,
- to identify a set of web pages dealing with the same theme,
- to identify a set of genes dedicated to the same function in the context of biological networks.
- ...

In a social network, we can deal with missing or modified information. In addition, the information exchanged can be often imperfect, due to the heterogeneous nature of the sources. In fact, information can be imprecise, uncertain or ambiguous:

An imprecise information is insufficient to answer questions of interest in a given situation. For example: Paul is between 20 and 25 years old.

An uncertain information is relative to the truth or the falsity of a proposition. For example: I believe Paul is 22 years old.

An ambiguous information can be noisy and interpreted in different ways. For example: Paul is young.

Several theories dealing with uncertainty exist in the literature such as the theories of probabilities, of possibilities and of belief functions. Historically, the formalism of probability theory is the most commonly used. Nevertheless, it does

not allow the modelling of ignorance. Indeed, in the absence of information, we associate the same probability with each event. In addition, due to the additivity axiom, the probability of an event implies a value on the probability of its complementary.

The limitations of this formalism was a motivation for the development of new theories of uncertainty such as the theory of possibilities and the theory of belief functions which impose no relation between an event and its complementary and it allow to easily model ignorance.

The theory of possibilities introduced by Zadeh (Zadeh, 1999) presents an alternative framework for representing uncertain information. This theory makes it possible to distinguish between plausible states and implausible ones. It uses fuzzy sets of mutually exclusive values.

To sum up, the theory of belief functions can be considered more general than that of probabilities or possibilities since we find these as particular cases. Indeed, if the mass is attributed to singletons only, the mass is called Bayesian mass function. In the case of attribution of the mass to nested focal elements, the mass is called consonant mass function.

Therefore, it would be interesting to model social network taking into consideration the fact that the information of the nodes, links and messages transiting on the social network can be imperfect.

In the same context, many studies focus on modeling the uncertain social network. In fact, they represent an uncertain network by weighting the nodes or links with values in $[0, 1]$ to model uncertainties. Hence, it will be easier to monitor the behaviour of the social network (Adar & Re, 2007).

In this thesis, we use the theory of belief functions (Dempster, 1967; Shafer, 1976) because it offers a mathematical framework for modelling uncertain and imprecise information. It has been employed in different fields, such as data classification (Denœux, 2008; Z.-G. Liu et al., 2015) and social network analysis (Wei et al., 2013).

Furthermore, the theory of belief functions provides a flexible way of combining information collected from different sources. In the majority of cases, this combination is followed by decision-making. It also allows conflict management.

In what follows, we will explain the aim of this thesis and will present the

outline of this work.

Aim and Scope

This thesis focuses on the problem of modelling interactions between nodes in a credibilist social network with special attention dedicated for community detection. The main research questions that have been addressed are as follows:

Advantage of using evidential attributes in social networks In the first contribution, we only consider attributes on the nodes and we aimed to answer the following questions:

First, how can we detect communities with uncertain attributes?

Second, to what extent the uncertain attributes make it possible to find the communities after adding noisy data?

In order to solve this problematic, we compared the clustering results of different type of uncertain attributes generated on the nodes based on the structure of the network. We consider two scenarios: random and sorted matrix of attributes.

Correction of noisy information in social networks using the belief function theory In the second contribution, the attributes on both nodes and links were considered. We were interested in solving the following issues:

How to classify nodes and links in their initial clusters in the presence of noisy data?

How to guarantee the coherence of the information of the network in the presence of noisy data?

Hence, to remedy to this problem, we propose an algorithm which allows the correction of the noisy information in the network based on the calculation of the distances between the triplets (a triplet contains 2 nodes and the link connecting them) composing the network and the coherent triplets defined initially based on the structure of the network.

Detection of spammed links in social networks using the belief function theory In the third contribution, we consider attributes on nodes, links and messages. We focused on finding a solution to the following problematic:

How, from the information on the nodes, links and messages, can we detect spammed links in social network?

In order to answer this question, we propose a method that aims to detect spammed links and take into account the imperfection of the information in the network. To do this, we assume that the class of a link can be changed depending on the type of messages that pass over it.

Thesis Outline

This thesis is organized as follows:

In Chapter 1, we will introduce some preliminary concepts related to social networks such as the definition of this latter in addition of its mathematical representation and its analysis. Then, we will detail some basic concepts of the theory of belief functions used in this thesis. After that, we will present some related works to the problem of classification in social network. Finally, we will introduce the used networks in the different processes of the experiments in this thesis.

In Chapter 2, we will detail our first contribution (Ben Dhaou et al., 2017) which consists of showing the advantage of using evidential attributes in social networks. Indeed, we will discuss how it allows to obtain better clustering results compared to other uncertain attributes.

In Chapter 3, we will present our second contribution (Ben Dhaou et al., 2018) which consists on the classification based on the structure of the network as well as on the attributes of both nodes and links. Indeed, we will detail the proposed algorithm which aims to correct the noisy information based on the calculation of the distances between the triplets of the network and the coherent ones defined initially.

In Chapter 4, we will explain how from the information on the nodes, links and messages, we are able to detect spammed links in social networks in the framework of belief functions. To do so, we will combine the mass functions of the links and messages at each iteration and then make a decision on the final class of

the link. Thus, we will determine if the link is spammed or not (Ben Dhaou et al., 2019).

Besides to theses chapters, we add the following appendices:

- Appendix A presents the LFR parameters used to generate the different networks.
- Appendix B introduces additional experiments of the first contribution.

Chapter 1

Social Networks

Nowadays, social networks represent a part of our everyday life. Indeed, they proposed new media to stimulate, accelerate and multiply the social interactions between individuals or groups. Not only in everyday life but also in a work-place, politics or in a media world, social networks have revolutionised relationships and linkages between the various components of society.

Social network analysis refers to relational theories that formalise social interactions in terms of nodes and links. The used concepts are from graph theory. Nodes are the social actors that interact with each other. However, they can also represent institutions. The links, as for them, are the relations between these nodes. There may be several types of links between nodes. In its simplest form, a social network is modelled to form an analysable structure where effective links between the nodes are studied.

In social networks, imperfect data can be observed. Indeed, these information can be imprecise, uncertain, ambiguous or even missing. In order to deal with this kind of situation, it has become necessary to use a theory that models and manages the imperfection of information such as the theory of belief functions. In fact, it offers a strong mathematical framework for modelling and managing the uncertainty in addition of a flexible way of combining information provided by different sources.

In this chapter, we start by defining and presenting the notion of social network as well as its representation and analysis in section 1.1. Next, we present some basic concepts of the theory of belief functions in section 1.2. Then, we

present the related works of the literature to this thesis in section 1.3 such as some community detections methods, some homophilic behaviour approach, link prediction researches and spammers detection methods. After that, we present the used social networks in this thesis in section 1.4.

1.1 Social Networks Representation and Analysis

In this section, we first define social network. Then, we present the way to represent it mathematically. Thereafter, we introduce the concept of social network analysis.

1.1.1 Social Network

A social network (Easley & Kleinberg, 2010) is a social structure made up of individuals (or organizations) called "nodes", which are tied (connected) by one or more specific types of interdependency, such as friendship, kinship, common interest, financial exchange, dislike, or relationships of beliefs, knowledge or prestige.

In this context, 3 categories of social networks have been defined as follows (Easley & Kleinberg, 2010):

The Web Social Networks allow to establish explicitly relations between users. Relationships in some of these websites are bilateral. This is for example the case of Facebook. In the case of Twitter or Youtube, the social relationship is established from unilateral way by the "follow" (following the publications from someone). Two individuals are neighbours in the associated graph if a relationship exists between the both.

Communication Networks are formed by transmissions of information between individuals. We find the same distinction between unilaterality (as for e-mails and SMS) and bilaterality (calls telephone and video-conferences). Two individuals are neighbours in the associated graph if they communicated using this network.

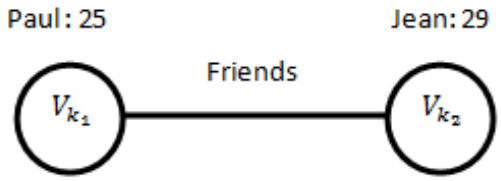

Figure 1.1: An Attributed Graph.

Collaborative Networks correspond to individuals who have worked together on a subject. For example, there are networks actors who have filmed together or scientists who have co-written articles. Relations in this type of network are bilateral. Two individuals are neighbours in the associated graph if they collaborated.

In what follows, we present how a social network is represented mathematically in addition of the definition of an attributed graph.

Social Network Representation A social graph is a representation of the interconnections among people, groups and organizations in a social network. A social graph helps to map the overall structure and interrelation of social network members. Mathematically, a social network is represented by a graph $G(V, E)$ where V is the set of nodes which represents persons, institutions, and so on and E is the set of edges which represents the type of relationships between the nodes (Mika, 2004).

Attributed Graphs According to (Seong et al., 1993), an attributed graph $G_a = (V_a, E_a)$ can be defined as a set of attributed vertices $V_a = \{v_1, \ldots, v_p, \ldots, v_q, \ldots, v_n\}$ and a set of attributed edges $E_a = \{\ldots, e_{pq}, \ldots\}$. The edge e_{pq} connects vertices v_p and v_q with an attributed relation.

Figure 1.1 represents an example of an attributed graph. The nodes have as attributes the name and the age of the person. As for the link, it has as attribute the type of the relationship between nodes.

1.1.2 Social Network Analysis

Social network analysis (Wasserman & Faust, 1994; Prell, 2012) has emerged as a key technique in modern sociology. It has also gained a significant following in

anthropology, biology, communication studies, economics, geography, information science, organizational studies, social psychology, sociolinguistics, and has become a popular topic of speculation and study.

In social network analysis (Wasserman & Faust, 1994; Prell, 2012) the observed attributes of social actors are understood in terms of patterns or structures of ties among the units. These ties may be any existing relationship between units; for example friendship, material transactions, etc.

Communities

Many networks of interest in the sciences, including social networks, computer networks, and metabolic and regulatory networks, are found to divide naturally into communities or modules for example, according to a center of interest, a category of age, a preference, etc.

Several researches gave different definitions of the term community. In the following, we present some of them.

In his work, Santo Fortunato (Fortunato, 2010) explained that communities represent groups of nodes which probably share common properties. He explains also that people naturally tend to form groups, within their work environment, family or friends.

Another definition is introduced in (Wasserman & Faust, 1994) as a set of nodes in which each of its subsets has more ties to its components within the set than outside.

(Radicchi et al., 2004) proposed the strong and weak definitions in order to relax the constraints of the LS-set. In a strong community, each node has more connections within the community than with the rest of the network. For the case of a weak community, the sum of all degrees within the community is larger than the sum of all degrees toward the rest of the network.

(Hu et al., 2008) define a community as a set of nodes and each node's degree inside the community should be larger than or at least equal to its degree link to any other community.

To sum up, a community represents a group of actors that share the same properties or interest and are more connected with entities inside the community

than the rest of the network.

Social Networks Properties

A social network has the following properties:

Local Preferential Attachment: This property (Leskovec et al., 2008) states that a node is more likely to create connections with vertices having a high degree and which are close.

Small World: It indicates that, going from neighbour to neighbour, it is possible to reach any another point of the graph in a small number of edges in average, even if the graph in question has a lot of nodes. It has been shown that real networks exhibit abundant short paths, notably the well-known "six-degrees of separation". Indeed, Milgram suggests that two people, randomly selected from American citizens, are connected on average by a chain of six relationships (Travers & Milgram, 1967; Amaral et al., 2000).

Community Structure: It appears when nodes can be grouped in a way such that vertices in a group are more connected to nodes in the same group compared to other vertices (Fortunato, 2010).

Community Homogeneity: This property takes place when the nodes inside a community are more similar according to their attribute values compared to nodes in a different community (Marsden, 1988).

An important sociological property in social networks is **homophily** (McPherson et al., 2001): Individuals know people who are similar to them. The structural consequence is that we observe many triangles, that is to say triplets of individuals forming 3-cliques. Indeed, if two individuals a and b know each other, and that c knows a but not b, c has the will and the opportunity to form a link with b:

- If a and b are similar, and a and c are similar, it is likely that c be similar to b.
- Because of the relationship of c with a, c has the opportunity to form a link with b.

This property also explains the appearance of highly connected sub-graphs in so-

cial networks.

Social graphs are **scale-free** (Barabási & Albert, 1999). We say that graph is scale-free if the degree distribution follows a Pareto law (called also power law), where the degree of a node is the number of nodes to which it is connected (γ is a positive number):

$$P(degree = k) \approx k^{-\gamma} \tag{1.1}$$

Some Metrics used in Social Network Analysis

Centrality This measure gives a rough indication of the social power of a node based on how well they "connect" the network. "Betweenness","Closeness", and "Degree" are all measures of centrality that will be presented afterwards.

Degree The degree of a node v is denoted $deg(v)$ and represents the count of the number of ties to other nodes in the network.

Betweenness The concept of betweenness centrality was introduced by (Freeman, 1978). It exhibits the extent to which a node lies between other nodes in the network. This measure takes into account the connectivity of the node's neighbours, giving a higher value for nodes which bridge clusters. The betweenness centrality of node v is given by the expression:

$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}} \tag{1.2}$$

Where σ_{st} is the total number of shortest paths from node s to node t and $\sigma_{st}(v)$ is the number of those paths that pass through v.

Closeness The closeness centrality was introduced by (Bavelas, 1950). It represents the degree an individual is near all other individuals in a network (directly or indirectly). It reflects the ability to access information through the "grapevine" of network members. Thus, closeness is the inverse of the sum of the shortest distances between each individual and every other person in the network. The

shortest path may also be known as the "geodesic distance". The closeness centrality is given by the following formula:

$$C(x) = \frac{1}{\sum_{y \in V} d(y, x)} \tag{1.3}$$

Where $d(y, x)$ is the geodesic distance between vertices x and y.

For large graphs, the closeness is given by:

$$C(x) = \frac{N}{\sum_{y} d(y, x)} \tag{1.4}$$

Where N is the number of nodes in the graph.

Clustering coefficient The clustering is a process that partitions a data set into homogeneous subclasses (clusters) so that the data in each subset share common characteristics.

Introduced in (Holland & Leinhardt, 1971; Watts & Strogatz, 1998), the clustering coefficient measures how close the neighbourhood of a vertex is. A higher clustering coefficient indicates a greater 'cliquishness'. We find two versions of the clustering coefficient measure: the global and the local.

The local clustering coefficient C_i for a vertex v_i is given by the proportion of links between the vertices within its neighbourhood divided by the number of links that could possibly exist between them.

The global clustering coefficient is given by the following equation:

$$C = \frac{Number\ of\ closed\ triplets}{Number\ of all\ triplets\ (open\ and\ closed)} \tag{1.5}$$

The number of closed triplets has also been referred to as 3× triangles in the literature (Holland & Leinhardt, 1971; Watts & Strogatz, 1998).

1.2 Uncertainty in Social Network

In this section, some basic concepts of the theory of belief functions are presented as well as some related works to the uncertainty in social networks.

1.2.1 Basics of the Theory of Belief Functions

Nowadays, several information from different sources are transiting on social networks. Most of the time, this information may be imperfect, imprecise, uncertain, vague or even incomplete. In order to manage the imperfections of the information, we choose to use the theory of belief functions. In fact, it can be considered more general than the theories of probabilities or possibilities since we find these as particular cases.

In this thesis, we choose to use the theory of belief functions because it is a powerful tool for representing imperfect information. In addition, it allows the combination of the information collected from different sources.

In what follows, we present some basic concepts of this theory.

The theory of belief functions (Dempster, 1967; Shafer, 1976) is a mathematical theory that extends probability theory by giving up the additivity constraint as well as the equal probability in the case of ignorance. Therefore, in probability theory equal probabilities do not distinguish equally probable events from the case of ignorance. In the theory of belief functions, cases of uncertainty, incompleteness and ignorance are modelled and distinguished. In this theory, justified degrees of support are assessed according to an evidential corpus. Evidential corpus is the set of all evidential pieces of evidence held by a source that justifies degrees of support awarded to some subsets.

Let $\Omega = \{\omega_1, \omega_2, ..., \omega_n\}$ be a finite and exhaustive set whose elements are mutually exclusive. The set Ω is called a frame of discernment, universe of discourse or domain of reference.

Let 2^Ω be a set of all subsets of Ω. It is made of hypotheses and unions of hypotheses from Ω. This set 2^Ω is called power set and defined as follows:

$$2^\Omega = \{A : A \subseteq \Omega\} \tag{1.6}$$

The mass function is a mapping from 2^Ω to $[0, 1]$ that allocates a degree of justified support over $[0, 1]$ to some subsets A of 2^Ω. The mass function is defined as follows:

$$m^\Omega : 2^\Omega \rightarrow [0, 1] \tag{1.7}$$

$$\sum_{A \subseteq \Omega} m^\Omega(A) = 1 \tag{1.8}$$

Every $A \in 2^\Omega$ such that $m^\Omega(A) > 1$ is called a focal element.

There are other functions related to mass functions which model differently the same piece of evidence and aim to simplify computations.

Credibility (or Belief) function

A credibility function, noted bel^Ω is the minimal degree of belief justified by available information. The credibility of a subset, $bel^\Omega(A)$, is the total belief on A. In order to compute the total belief on A, the masses of proper subsets B of A, $m^\Omega(B)$, must be summed to $m^\Omega(A)$. Therefore, $bel^\Omega(A)$ is obtained by summing masses of subsets of A. The credibility function is given by:

$$bel^\Omega : 2^\Omega \rightarrow [0, 1] \tag{1.9}$$

$$bel^\Omega(A) = \sum_{B \subseteq A, B \neq \emptyset} m^\Omega(B) \tag{1.10}$$

It should be noted that the mass function that produces a given credibility function is unique and thus can be recovered from the credibility function as follows:

$$\begin{cases} m^\Omega(A) = \sum_{\emptyset \neq B \subseteq A} (-1)^{|A|-|B|} bel^\Omega(B) \;\; \forall A \subset \Omega, A \neq \emptyset \\ m^\Omega(\emptyset) = 1 - bel^\Omega(\Omega) \end{cases} \tag{1.11}$$

$\overline{A}$ is the complement of A in Ω. As the empty set is included in both A and $\overline{A}$, it is discarded from the sum.

Plausibility function

The plausibility function, noted pl^Ω, is the maximum amount of potential support that could be given to a subset A. It is measured by summing masses of propositions compatible with A. The plausibility function is defined as follows:

$$pl^\Omega : 2^\Omega \rightarrow [0, 1] \tag{1.12}$$

$$pl^\Omega(A) = \sum_{A \cap B \neq \emptyset, B \subseteq \Omega} m^\Omega(B) \tag{1.13}$$

Functions *Bel* and *Pl* are linked by the following relation:

$$pl^{\Omega}(A) = 1 - bel^{\Omega}(\overline{A}) \tag{1.14}$$

The mass function that produces a given plausibility function is unique and thus can be recovered using the following equations:

$$\begin{cases} m^{\Omega}(A) = \sum_{A \subseteq B} (-1)^{|B|-|A|+1} pl^{\Omega}(\overline{A}) \\ m^{\Omega}(\emptyset) = 1 - pl^{\Omega}(\Omega) \end{cases} \tag{1.15}$$

It should be noted that under the closed world assumption, $m^{\Omega}(\emptyset) = 0$ and $bel^{\Omega}(\Omega) = pl^{\Omega}(\Omega) = 1$. However, with the open world assumption, the mass function $m^{\Omega}(\emptyset)$ can be viewed as missing mass or a not committed mass equal to $1 - pl^{\Omega}(\Omega)$.

Some particular mass functions are introduced in the following:

Categorical mass functions

A categorical mass function is a normalized mass function which has a unique focal element A^*. This mass function is noted $m^{\Omega}_{A^*}$ and defined as follows:

$$m^{\Omega}_{A^*}(A) = \begin{cases} 1 \text{ if } A = A^* \subset \Omega \\ 0 \ \forall A \subseteq \Omega \text{ and } A \neq A^* \end{cases} \tag{1.16}$$

Vacuous mass functions

A vacuous mass function is a particular categorical mass function focused on Ω. It means that a vacuous mass function is normalized and has a unique focal element which is Ω. This type of mass function is defined as follows:

$$m^{\Omega}_{\Omega}(A) = \begin{cases} 1 \text{ if } A = \Omega \\ 0 \text{ otherwise} \end{cases} \tag{1.17}$$

Vacuous mass function emphasizes the case of total ignorance.

Dogmatic mass functions

A dogmatic mass function is a mass function where Ω is not a focal element. A dogmatic mass function is defined as follows:

$$m^{\Omega}(\Omega) = 0 \tag{1.18}$$

Bayesian mass functions

A Bayesian mass function is a mass function which all focal elements are elementary hypotheses. It is defined as follows:

$$\begin{cases} m^{\Omega}(A) \in [0,1] \text{ if } |A| = 1 \\ m^{\Omega}(A) = 0 \text{ otherwise} \end{cases} \tag{1.19}$$

As all focal elements are single points, this mass function is a probability distribution.

Consonant mass functions

A consonant mass function is a mass function which focal elements are nested ($A_1 \subset A_2 \subset ... \subset \Omega$).

Certain mass functions

A certain mass function is a categorical mass function such that the only focal element is an elementary hypothesis. This mass function emphasizes the case of total certainty as the source supports only one hypothesis with certainty. It is defined as follows:

$$m^{\Omega}(A) = \begin{cases} 1 \text{ if } A = \omega \in \Omega \\ 0 \ \forall A \subseteq \Omega \text{ and } A \neq \omega \end{cases} \tag{1.20}$$

Simple support functions

A simple support function is a mass function which has only one focal element other than the frame of discernment Ω. This unique focal element is called the

focus of the simple support function. A simple support function is defined in (Shafer, 1976; Smets, 1995) as follows:

$$m^{\Omega}(B) = \begin{cases} \omega \text{ if } B = \Omega \\ 1 - \omega \text{ if } B = A \text{ for some } A \subset \Omega \\ 0 \text{ otherwise} \end{cases} \tag{1.21}$$

where A is the focus of the simple support function and $\omega \in [0, 1]$.

A simple support function is also noted A^{ω} where ω is the degree of support of the frame of discernment Ω and the complement of ω to 1 is the degree of support of the focus A.

The simple support function is used to represent uncertainty, imprecision and ignorance.

Example

Suppose the frame of discernment of the nodes $\Omega_N = \{C_1, C_2, C_3\}$ and assume a support mass function m^{Ω_N} defined on Ω_N: $m^{\Omega_N}(C_1 \cup C_3) = 0.7$, $m^{\Omega_N}(\Omega_N) = 0.3$ This function models both the uncertainty (0.7) and the imprecision on $C_1 \cup C_3$.

Consistent mass functions

A consistent mass function is a function which all focal elements have a non empty intersection. For such mass functions, at least one focal element is common to all the focal ones.

Pignistic Transformation

In the credal level, degrees of belief are assessed and mass functions can be combined. In the pignistic level, decisions are made according to criteria. It consists on choosing the most probable hypothesis from Ω. Based on the method proposed by Smets (Smets, 2005), each mass of belief $m(A)$ is equally distributed among the elements of A. This leads to the concept of pignistic probability, $BetP$, defined by:

$$BetP_m^{\Omega}(\omega_i) = \sum_{\omega_i \in A, A \subseteq \Omega} \frac{m^{\Omega}(A)}{|A|(1 - m^{\Omega}(\emptyset))} \tag{1.22}$$

Decision is made according to the maximum of pignistic probabilities.

Product Space

In some applications, pieces of evidence may be defined on different frames of discernment. To assess flexibly justified degrees of support in different frames, some tools provide the redefinition of these pieces under a common space. Suppose that we have two different frames of discernment $\Omega = \{\omega_1, \omega_2, ..., \omega_{n_1}\}$ and $\Theta = \{\theta_1, \theta_2, ..., \theta_{n_2}\}$. The frame of discernment $\Omega \times \Theta$ is composed of the Cartesian product of Ω and Θ, $\Omega \times \Theta$ is defined as follows:

$$\Omega \times \Theta = \{(\omega_1, \theta_1), (\omega_2, \theta_2), ..., (\omega_1, \Theta), ..., (\Omega, \Theta)\} \tag{1.23}$$

Example

Let Ω_N the frame of discernment of the nodes and Ω_L the frame of discernment of the links in a given social network:

- $\Omega_N = \{C_1, C_2, C_3\}$. The attributes C_i represent to which community the nodes belong.
- $\Omega_L = \{IC_1, IC_2, IC_3\}$. The attributes IC_i represent to which community the links belong

The product frame is as follows:
$\Omega_N \times \Omega_L = \{(C_1, IC_1), (C_1, IC_2), (C_1, IC_3), (C_2, IC_1), (C_2, IC_2), (C_2, IC_3), (C_3, IC_1), (C_3, IC_2), (C_3, IC_3)\}$

Multivalued Mapping

To focus on the type of relationship between two different frames of discernment Ω and Θ, we may use the multivalued mapping introduced by Hyun Lee (H. Lee, 2011):

$$m^{\Theta}_{\Gamma}(B_j) = \sum_{\Gamma(e_i)=B_j} m^{\Omega}(e_i) \tag{1.24}$$

with $e_i \subseteq \Omega$ and $B_j \subseteq \Theta$. Therefore the function Γ is defined as follows:

$$\Gamma : \Omega \rightarrow 2^{\Theta} \tag{1.25}$$

Vacuous Extension

The vacuous extension (Smets, 1993) is a tool to extend a mass function defined on a frame of discernment Ω (*or* Θ) to the product frame $\Omega \times \Theta$. The vacuous extension, noted $\uparrow$, consists on a transfer of basic belief masses of each focal element B to its cylindrical extension ($B \times \Theta$ is the cylindrical extension of B) as follows:

$$m^{\Omega\uparrow\Omega\times\Theta}(A) = \begin{cases} m^{\Omega}(B) & \text{if } A = B \times \Theta, B \subseteq \Omega \\ 0 & \text{otherwise} \end{cases} \tag{1.26}$$

The vacuous extension is a particular case of the multivalued mapping operation.

Example Suppose that we have the following mass functions:

$m^{\Omega_N}(C_1) = 0.5$ and $m^{\Omega_N}(\Omega_N) = 0.5$

To extend m^{Ω_N} from Ω_N to $\Omega_N \times \Omega_L$, the mass of each focal element is transferred to its cylindrical extension. Thus, we obtain:

- $m^{\Omega_N\uparrow\Omega_N\times\Omega_L}(C_1, \Omega_L) = 0.5$
- and $m^{\Omega_N\uparrow\Omega_N\times\Omega_L}(\Omega_N, \Omega_L) = 0.5$

Distance of Jousselme

The distance of Jousselme (Jousselme et al., 2001) represents the degree of similarity between bodies of evidence. It is defined by:

$$d_j(m_1^{\Omega}, m_2^{\Omega}) = \sqrt{\frac{1}{2}(m_1^{\Omega} - m_2^{\Omega})^T \mathbf{Jac}(m_1^{\Omega} - m_2^{\Omega})} \tag{1.27}$$

where the elements $Jac(A, B)$ of Jaccards weighting matrix **Jac** are defined as:

$$Jac(A, B) = \begin{cases} 1 \text{ if } A = B = \emptyset \\ \dfrac{|A \cap B|}{|A \cup B|}, \text{otherwise} \end{cases} \tag{1.28}$$

Combining efficiently several mass functions coming from distinct sources represents a major information fusion problem in the theory of belief functions. Many

rules have been proposed in (Dempster, 1967; Yager, 1987; Dubois & Prade, 1988; Smets, 1990; Denceux, 2006; Martin & Osswald, 2007)

In the following, we recall some popular combination rules.

Conjunctive combination rule

Proposed by Smets (Smets, 1990), the conjunctive combination rule allows to associate a positive mass to the empty set. It is interpreted as the non exhaustivity of the frame of discernment. The conjunctive combination rule for two mass functions m_1^Ω and m_2^Ω is defined as follows:

$$m_{1\textcircled{\cap}2}^\Omega(A) = \sum_{B\cap C=A} m_1^\Omega(B) \times m_2^\Omega(C) \tag{1.29}$$

Dempster combination rule

The Dempster combination rule (Dempster, 1967) is a normalized conjunctive rule. Given for two mass functions m_1^Ω and m_2^Ω for all $X \in 2^\Omega$, $X \neq \emptyset$, it is defined by:

$$m_\oplus^\Omega(X) = \frac{1}{1-k} \sum_{A\cap B=X} m_1^\Omega(A).m_2^\Omega(B) \tag{1.30}$$

where $k = \sum_{A\cap B=\emptyset} m_1^\Omega(A).m_2^\Omega(B)$ is the global conflict of the combination. This rule is adapted when the combined mass functions are cognitively independent.

The cautious combination rule

The cautious combination rule (Denceux, 2006) of two mass functions m_1^Ω and m_2^Ω issued from dependent sources is defined as follows:

$$m_1^\Omega \textcircled{\wedge} m_2^\Omega = \textcircled{\cap}_{A\subset\Omega} A^{w_1(A)\wedge w_2(A)} \tag{1.31}$$

Where $A^{w_1(A)}$ and $A^{w_2(A)}$ are simple support functions focused on A with weights w_1 and w_2. $\wedge$ represents the min operator of simple support functions weights. When the min operator $\wedge$ is replaced by the max operator $\vee$, the bold combination rule is obtained.

Both cautious rules are used to combine mass functions issued from dependent sources. In addition, they are commutative, associative and idempotent.

Mean combination rule

The mean combination rule, m^{Ω}_{Mean}, of two mass functions m^{Ω}_1 and m^{Ω}_2 is the average of these ones. Therefore, for each focal element A of n mass functions, the combined one is given by:

$$m^{\Omega}_{Mean}(A) = \frac{1}{n}\sum_{i=1}^{n} m^{\Omega}_i(A) \tag{1.32}$$

This rule can be used in the case of dependent mass functions.

PCR6 combination rule

The *PCR*6 combination rule proposed by (Martin & Osswald, 2007) is dedicated to combine two or many mass functions. For two mass functions, the *PCR*6 is defined by:

$$m_{PCR6}(A) = m_{1\bigcirc\!\!\!\!\cap 2}(A) + \sum_{B\in 2^{\Omega}, A\cap B=\emptyset}\left(\frac{m_1(A)^2 m_2(B)}{m_1(A)+m_2(B)} + \frac{m_2(A)^2 m_1(B)}{m_2(A)+m_1(B)}\right) \tag{1.33}$$

When it comes to the combination of M mass functions provided by M independent and distinct sources, the authors proposed a generalised combination rule in (Martin & Osswald, 2006).

1.2.2 Related Works to the Uncertainty in Social Networks

Several researches were focused on managing the imperfections of the information in a social network in order to remedy problems such as detection of spammers, link prediction,etc.

To do so, many authors opted to combine the theory of graphs with the theories dealing with uncertainty like probability (Khan et al., 2014; Parchas et al., 2014), possibility or theory of belief functions (Ben Dhaou et al., 2014) in order

to provide a general framework for an intuitive and clear graphical representation of real-world problems.

Indeed, in their work, (Khan et al., 2014) studied reliability search on uncertain graphs (also called probabilistic graphs). The authors proposed a novel index RQ-tree which is based on hierarchical clustering of the nodes in the graph, and further optimized using a balanced-minimum-cut criterion.

As for (Parchas et al., 2014), they proposed algorithms for creating deterministic representative instances of uncertain graphs that maintain the underlying graph properties. Specifically, the algorithms aim to preserve the expected vertex degrees because they capture well the graph topology.

In the same context, we introduced in (Ben Dhaou et al., 2014) a belief social network. The purpose of this work is to model a social network as being a network of fusion of information and determine the true nature of the received message in a well-defined node.

A belief social network is represented by associating a mass function to each node and link. Formally, an evidential graph $G = \{V^b, E^b\}$ is composed of a set of nodes V^b and a set of edges E^b. A mass $m^{\Omega_N}_{V_i}$ defined on the frame of discernment Ω_N of the nodes is associated to every node V^b_i of V^b. Moreover, a mass $m^{\Omega_L}_{V_{ij}}$ defined on the frame of discernment Ω_L of the edges is attributed to every edge (V^b_i, V^b_j) of E^b. Then, a mass function is associated to each message transiting in the network .

In order to determine the true nature of the message in a defined node, the information of the node and link are combined to obtain a belief of the network. After that, an operation to transfer the information of the network defined on the product space $\Omega_N \times \Omega_L$ to the frame of discernment of the messages Ω_M is used. Next, the obtained mass function of the message based on the information of the network is combined with the initial one. Finally, the decision on the nature of the resulting message is taken by using the pignistic probability.

We presented in this section some basic concepts of the theory of belief functions as well as related works to the uncertainty in social networks. In this thesis, we use the simple support function for the generation of the attributes of the nodes, links and messages. The distance of Jousselme is used to compute the degree of similarities between the mass functions. In order to combine the information from different frames of discernment, we use the notion of the product space, the

vacuous extension and the multi-valued mapping operation. Then, to combine information obtained from different independent sources, we use the Dempster combination rule. In the case of dependent sources, we use the Mean combination rule. Finally, to make decision, we use the pignistic probability.

The belief social network introduced by (Ben Dhaou et al., 2014) will be used as a representation of social networks with evidential attributes in all the proposed contributions.

We present in the next section some related works to the community detection in social networks.

1.3 Clustering in Social Networks

Clustering is the assignment of a set of observations to subsets called clusters so that the observations in the same cluster share similarities. Clustering is an unsupervised learning method and a common technique of statistical data analysis used in many fields such as community detection in social networks. Indeed, in social networks, we aim to find groups of individuals with dense links internally and sparse links externally.

In this section we present related works to the community detection task. We start by presenting some community detection methods taking into account the graph structure only and some other taking into consideration the attributes in addition of the graph structure. Then, we present few approaches dealing with the homophilic behaviour in social networks. After that, we recall the principle of the K-Medoids algorithm. Next, we introduce some researches that have focused on the link prediction problems. Thereafter, we present some recent works dealing with the spammers detection problem. Finally, we recall the principle of the Walktrap approach.

1.3.1 Some Community Detection Methods with only Graphs Structures

Many researches aim to find communities based on the network structure. In the following, we introduce some of them. In the literature, there are several

studies such as the hierarchical clustering (Scott, 2017) which is a method based on the development of a measure of similarity between pairs of vertices using the network structure. The disadvantage of this technique consists on ignoring the number of communities that should be used to get the best division of the network.

The second type of methods is the algorithms based on edge removal. Two techniques are presented:

The algorithm of Girvan and Newman (Girvan & Newman, 2002) which is a divisive method, in which edges are progressively removed from a network. In addition, the edges to be removed are chosen by computing the betweenness scores. The final step consists on recomputing the betweenness scores following the removal of each edge. The betweenness of an edge is defined to be the number of geodesic paths between node pairs that run along the edge in question, summed over all node pairs. The proposed approach involves simply calculating the betweenness of all edges in the network, removing the one with highest betweenness and repeating this process until no edges remain. In the case where two or more edges tie for highest betweenness, one can either choose one at random to remove, or simultaneously remove all of them. This algorithm does not provide any guide to how many communities a network should split into. In addition, it is also slow. In order to address the first issue, the authors propose in (Newman & Girvan, 2004) that the generated divisions should be evaluated using a measure they call modularity, which is a numerical index of how good a particular division is. In (Newman, 2004), the author proposed a new algorithm for extracting community structure based on the notion of modularity which allows to address the second issue. Indeed, the new algorithm is much faster than the previous ones.

The algorithm of (Radicchi et al., 2004) is also based on iterative removal of edges. The authors show the way to implement in practice in the existing algorithms the quantitative definitions of community. In addition, they propose a local algorithm to detect communities which is performing better than the existing algorithms with respect to computational cost and keeping the same level of reliability. In this work, the authors introduce a general criterion for deciding which of the sub-graphs singled out by the detection algorithms are actual communities. In addition, they present an alternative algorithm based on the computation of local quantities. It gives results similar to the Girvan and Newman approach in controlled cases. However, it is much better from the point of view of computational speed.

All the methods cited above focused only on the structure of the network and do not take into account the nodes attributes. In fact, often, nodes have features associated with them.

1.3.2 Some Community Detection Methods with Graphs Structure and Attributes

We present in the following some community detection methods based on graph structure and attributes.

The presented model in (Y. Zhou et al., 2009) uses both structure and attributes. First, the authors performed a unified neighbourhood random walk distance measure which allows to measure the closeness of vertex on an attribute augmented graph. Then, the authors use a K-Medoids clustering method to partition the network into k clusters. The authors propose an unified distance measure in order to combine structural and attribute similarities. Second, they provide a theoretical analysis to quantify the contribution of attribute similarity to the unified random walk distances to measure node closeness. Third, the authors propose a weight self-adjustment method in order to learn the degree of contributions of different attributes in random walk distances. In addition, they prove that the edge weights are adjusted towards the direction of clustering convergence. Finally, the authors test their approach in real large graphs. The experiments show that the proposed method is able to partition the graph into high-quality clusters with cohesive structures and homogeneous attributes values. The clustering algorithm converges very quickly.

A second method presented in (Leskovec & Mcauley, 2012) consists on a model dedicated to detect circles that combines network structure and user profile. The circle represent a group of persons connected to each user. On Facebook, it is called ”list” of friends and ”circles” on Google+. The authors learn for each circle, its members and the circle-specific user profile similarity metric. From the modelling of node membership to multiple circles, the method proposed by the authors is able to detect overlapping as well as hierarchically nested circles. In their work, the authors model circle affiliation as latent variables and similarity between alters as a function of common profile information. The proposed approach is unsupervised and aims to learn which dimensions of profile similarity lead to densely linked circles. In addition, the method is able to predict hard as-

signment of a node to multiple circles and to learn the dimensions of similarity along which links emerge.

A third method presented in (Trabelsi et al., 2016) consists on dealing with the uncertainty that occurs in the attribute values within the belief functions framework in the case of clustering. The authors develop another version of decision trees using the theory of belief functions and are interested in the case where the uncertainty occurs in both construction and classification phases. In their paper, the authors present a new decision tree composed of two procedures. The first one consists on the construction of the tree from containing uncertain attributes. The second one consists on the classification of new instances described by uncertain attribute values. In this work, the time complexity still a critical problem, especially for large or even medium sized databases.

The works cited above (Y. Zhou et al., 2009; Leskovec & Mcauley, 2012) use only a probabilistic attributes as well as the structure of the graph to do the clustering. In our previous work (Ben Dhaou et al., 2017), we show that the use of evidential attributes gives better results than the probabilistic ones in the clustering.

The works presented in (Y. Zhou et al., 2009; Leskovec & Mcauley, 2012; Trabelsi et al., 2016) are interesting, but they do not assume that network information can be noisy or perturbed. In addition, they do not consider the use of node and link attributes simultaneously to do the clustering.

1.3.3 Homophilic Behaviours in Social Networks

In (K. Zhou, Martin, Pan, & Liu, 2018), the authors present a new method using the theory of belief functions that aims to detect communities on graphs after the stabilisation of the label propagation process. In fact, SELP (Semi-supervised clustering approach based on an Evidential Label Propagation strategy) permits to propagate the labels from the labelled nodes to the unlabelled ones based on a propagation rule. The proposed algorithm computes the dissimilarities between nodes based on the graph structure. The main advantage of the proposed algorithm is that it can effectively use limited supervised information to guide the process of the detection.

Another interesting work presented in (Guimerà & Sales-Pardo, 2009) aims to identify missing and spurious interactions (links connecting nodes) and to recon-

struct network whose properties are closer to the 'true' underlying network. To do so, the authors focus on the family of stochastic block models. The proposed method can also guide new discoveries. In fact, if a given interaction between 2 nodes exists but with a very low reliability for the interaction, that means that the function of the interaction is very specific.

The method proposed in (Vuokko & Terzi, 2010) aims to address the problem of reconstructing the original network and set of features given their randomized counterparts. The technique of data randomization consists of removing some of the original edges of the network in addition of new ones. Furthermore, the features can be also randomized. In this work, the authors assume that data-randomization method does not completely destroy the original dataset. For the case of features, every node is associated with k binary features. If the node has that feature, it will take 1 otherwise it will take 0.

All the works presented are interesting. However, we can not do a comparison at the experimental level since we do not consider the resolution of the same problem which is the correction of noisy information in social network. Indeed, the first work consider a network with few nodes having labels and aim to propagate them to the unlabelled ones. In this thesis, all nodes and links have a prior labels. In the second research (Guimerà & Sales-Pardo, 2009), the authors are interested in predicting links based on observations. In this work, the initial structure of the network is not modified. Regarding the third work (Vuokko & Terzi, 2010), the authors remove links from the graph and add new ones whereas in our case, the structure of the graph is not amended.

1.3.4 K-Medoids algorithm

The K-medoids algorithm (Arora et al., 2016) is a partitional clustering algorithm: It breaks the dataset up into groups. The K-medoids algorithm minimizes the sum of dissimilarities between points labelled to be in a cluster and a point designated as the centre of that cluster. Unlike the K-means algorithm, K-medoids algorithm chooses data-points as centres called medoids.

A medoid algorithm of a finite dataset is a data point from this set, whose average dissimilarity to all the data points is minimal.

K-medoids is a partitioning technique of clustering that clusters the dataset of n objects into k clusters with k known a priori. Among the strengths of K-medoids,

we mention the fact that it is more robust to noise and outliers comparing to K-means. Indeed, it minimizes the sum of general pairwise dissimilarities.

The basic idea of the K-medoids algorithm is to first compute the K representative objects which are called as medoids. After finding the set of medoids, each object of the data set is assigned to the nearest medoid: object i is put into cluster K_i, when medoid mK_i is nearer than any other medoid m_w.

Among the K-medoids methods proposed in the literature, we mention the method "PAM: Partitioning around Medoids", the method "CLARA: Clustering LARge Applications" introduced by (Kaufman & Rousseeuw, 2009), the method "CLARANS: Clustering Large Applications based upon RANdomized Search" (Ng & Han, 2002) and the method "ECMdd: Evidential C-Medoids" proposed by (K. Zhou et al., 2016).

Given the advantages of this algorithm, we use it in our first contribution to the partitioning of the nodes of the different used networks.

1.3.5 Link Prediction Problem in Social Networks

Several works have focused on the problem of prediction of links in social networks. Indeed, social networks are highly dynamic objects; they grow and change quickly over time through the addition of new edges, signifying the appearance of new interactions in the underlying social structure. Therefore, the link prediction problem becomes an important task which aims for predicting the likelihood of a future association between two nodes, knowing that there is no association between the nodes in the current state of the graph.

The authors in (Al Hasan & Zaki, 2011) present a survey of some representative links prediction methods by categorising them by the type of the models: the traditional models which extract a set of features to train a binary classification mode. The second type of methods is the probabilistic approaches which model the joint-probability among the entities in a network by Bayesian graphical models. Finally, the linear algebraic approach which computes the similarity between the nodes in a network by rank-reduced similarity matrices.

Other authors have been interested in the problem of predicting links by considering it in an uncertain context such as the presented work in (Mallek et al., 2015). Indeed, the authors examined the link prediction problem by adopting the

theory of belief functions. In their work, they proposed a new graph-based model for social networks that encapsulates the uncertainties in the links structures. In addition, they used the assets of the theory of belief functions for combining pieces of evidence induced from different sources and decision making in order to propose a novel approach for predicting future links through information fusion of the neighbouring nodes.

Another interesting research proposed by (Moradabadi & Meybodi, 2017) deals with the problem of links prediction in the case of fuzzy social networks based on distributed learning automata (FLP-DLA). Indeed, the authors started by modelling the social network as a fuzzy social network, where each link has a fuzzy strength. The fuzzy strength is defined using the date of link occurrence and the number of collaborations in the corresponding link, since it is assumed that very old links are not important in the prediction task. After that, the fuzzy links are used by distribution learning automata to find the strength of a path for any link that must be predicted. DLA tries to find the path strength using a reinforcement mechanism and graph navigation.

All these works are interesting. However, the cited researches focus only on how to add links to the network when an entity disappears.

1.3.6 Detection of Spammers in Social Networks

Social networks are extremely popular among Internet users. Indeed, users spend a significant amount of time storing and sharing personal information. Unfortunately, this kind of information attracts the interest of cybercriminals. These latter might use it in order to identify theft or to drive target spam campaigns. In addition, spammers can exploit the relationships between users with the intention of luring victims to malicious websites. According to (Washha et al., 2016), a spammer is a goal-oriented person who aims to achieve unethical goals. A spammer proceeds by exploiting trending topics to lunch its spammy content. In this context, several researches have focused on the analysis and detection of spammers in social networks.

In order to remedy this problem, the authors of (K. Lee et al., 2010) proposed a honeypot-based approach for uncovering social spammers in on-line social networks. The purpose of their method is to automatically harvest spam profiles from social networking communities avoiding the drawbacks of burdensome human in-

spection. In addition, the authors aim for developing robust statistical user models to distinguish between social spammers and legitimate users.

In (Stringhini et al., 2010), the authors analyse to which extent spam has entered social networks. Indeed, they analyse how spammers who target social networking sites operate. To do so, the authors created a set of honeynet accounts on 3 major social networks and logged all the activity observed by these profiles. Then, they investigate how spammers are using social networks and examine the effectiveness of the counter-measures implemented by the major social network portals to prevent spamming on their platforms. Based on this information, the authors identify characteristics that allow them to detect spammers in a social networks.

In (Z. Yang et al., 2014) the authors used ground-truth data about the behaviour of Sybils in the wild in order to create a measurement-based, real-time Sybil detector. In addition, they characterised the Sybil graph topology on a major on-line social network. The authors analysed also the behaviour of Sybil clickstream on Renren. Indeed, their data captures the exact session-level sequences of actions that Sybils use to send spam and generate friend requests.

The authors of (Zheng et al., 2015) adopt the spammers features to detect spammers and test the result over Sina Weibo. In addition, they study a set of most important features related to message content and user behaviour in order to apply them on the SVM (Support Vector Machine) based classification algorithm for spammer detection.

Although the proposed approach could achieve precise classification result, it takes over an hour in a process for model training. Furthermore, in the era of big data with huge data volume and convenient access, feature extraction mechanism in the proposed model might be low adaptive and take a lot of time.

Another interesting work proposed in (Roul et al., 2016) consists of detecting spam web pages by using either content or link-based techniques or combination of both. In the content-based approach, the authors used term density and Part of Speech (POS) ratio test in order to detect the spam pages. In the link-based method, they used collaborative detection using personalised page ranking to detect spam pages. As future works, the authors intend to extend their model by finding topical spam patterns to understand different tricks played by spammer in different web pages. In addition, in order to reduce the running time of the algorithm, they intend to work in a distributed environment using map-reduce such as

Hadoop.

The authors in (Martinez-Romo & Araujo, 2013) introduced a method based on the detection of spam tweets in isolation and without previous information of the user and the application of a statistical analysis of language to detect spam in trending topics. The authors present an approach to detect spam tweets in real time using language as the primary tool.

Although the work presented is interesting, the analysed dataset is limited and may still contain some bias. In addition, the number of spam tweets is a lower bound of the real number. As a future work, the authors intend to select the most appropriate features for use in a detection system in real time.

In (Washha et al., 2016), the authors present an approach for detecting spammers on Twitter. In their work, they try first to find to what extent it is possible to increase the robustness of user's and content features used in the literature. Then, the authors were interested to sort out if there is an accessible and unmodifiable property overtime such that it can be leveraged for advancing the available features as well as designing new features.

To sum up, some works in the literature focused on the prediction of the class label of tweet such as in (Martinez-Romo & Araujo, 2013). Other researches (Washha et al., 2016; Zheng et al., 2015) were interested on analysing the user's profile to predict whether the user is a spammer or not. Thus, the possibility that the link can be spammed is not taken into consideration.

1.3.7 Walktrap Approach

In (Pons & Latapy, 2005), the authors introduced a measure of similarities between vertices based on random walks. These latter tend to get trapped into densely connected sub-graphs corresponding to communities. Based on some properties of random walks in graphs, the authors presented a distance of the structural similarity between nodes and between communities.

In their work, (Pons & Latapy, 2005) proposed first a distance r between nodes that capture the community structure of the graph. It is defined by:

$$r_{ij}(t) = \sqrt{\sum_{k=1}^{n} \frac{(P_{ik}^t - P_{jk}^t)^2}{d(k)}} = \|D^{-\frac{1}{2}} P_i^t - D^{-\frac{1}{2}} P_j^t\| \tag{1.34}$$

where i and j two nodes of the graph, $\|.\|$ is the Euclidean norm of $\mathbb{R}^n$, $P^t_{i.}$, $P^t_{j.}$ are 2 probability distributions, D is the diagonal matrix of the degrees and $d(k)$ is the degree of the node k.

- If two nodes i and j are in the same community, the probability P^t_{ij} will be hight. However, it does not necessarily imply that i and j are in the same community.
- The probability P^t_{ij} is influenced by the degree $d(j)$. This is due to the fact that the walker has higher probability to go to nodes having high degree.
- Two nodes of same community tend to "see" all the other nodes in the same way. Therefore, if i and j are in the same community, we will probably have $\forall k, P^t_{ik} \simeq P^t_{jk}$

Then, they introduced a second distance between two communities C_1 and C_2 which is given by the following equation:

$$r_{C_1C_2}(t) = \sqrt{\sum_{k=1}^{n} \frac{(P^t_{C_1k} - P^t_{C_2k})^2}{d(k)}} = \|D^{-\frac{1}{2}}P^t_{C_1} - D^{-\frac{1}{2}}P^t_{C_2}\| \tag{1.35}$$

This method is used in the next chapter in order to compare the obtained results given by our first contribution with those given by the Walktrap approach.

1.3.8 Clustering Results Evaluation Metrics

In what follows, we present some metrics used to evaluate the clustering results such as the normalized mutual information, the variation of information, the rand index and the adjusted rand index.

Let X be a finite set with cardinality $|X| = n$ and C and C' two clustering algorithms of X.

NMI: Normalized Mutual Information The Normalized Mutual Information (Knops et al., 2006) measures the similarity between the planted partitions (ground

truth) and the clustering results given by the algorithms. It measures the proportion of the nodes that have been grouped correctly and represents the consistency between the found community structure and the presumed one. The NMI is defined between 0 (completely different clusterings) and 1 (identical clusterings).

The NMI is given by:

$$NMI(A,B) = \frac{\mathcal{H}(A)+\mathcal{H}(B)}{\mathcal{H}(A,B)} \tag{1.36}$$

with $\mathcal{H}$ the entropy given by:

$$\mathcal{H}(A) = -\sum_{a\in A} P_A(a)\log P_A(a) \tag{1.37}$$

$$\mathcal{H}(A,B) = -\sum_{a\in A, b\in B} P_{A,B}(a,b)\log P_{A,B}(a,b) \tag{1.38}$$

Where A and B are two discrete random variables. The NMI effectively measures the amount of statistical information shared by the random variables representing the cluster assignments and the user-labelled class assignments of the data instances.

VI: Variation of Information The variation of Information between two clustering algorithms introduced by Meilă (Meilă, 2003) is a measure based on the entropy. It is defined by:

$$\mathcal{VI}(C,C') = \mathcal{H}(C)+\mathcal{H}(C')-2\mathcal{I}(C,C') \tag{1.39}$$

$$= [\mathcal{H}(C-\mathcal{I}(C,C')]+[\mathcal{H}(C')-\mathcal{I}(C,C')] \tag{1.40}$$

with $\mathcal{VI}(C,C')$ represents the amount of information about C that we loose and the second term of the equation corresponds to the amount of information about C' that we still have to gain.

The variation of information is not bounded by a constant value. However there is an upper bound equal to $2logK$ with K is the number of clusters. Thus, the more the result is similar to the benchmark, the smaller the value of $\mathcal{VI}$ is. If the value of $\mathcal{VI}$ is equal to 0, it means that we have identical clustering algorithms.

Rand Index (Rand, 1971) Instead of counting single elements, the Rand index counts correctly classified pairs of elements. It is defined by:

$$\mathcal{R}(C, C') = \frac{2(n_{11} + n_{00})}{n(n-1)} \tag{1.41}$$

Where:

- n_{11} = { pairs that are in the same cluster under C and C' }
- n_{00} = { pairs that are in different clusters under C and C' }
- n is the cardinality of the set X.

The Rand index $\mathcal{R}$ ranges from 0 (no pair classified in the same way under both clustering algorithms) to 1 (identical clustering). It should be noted that the value of $\mathcal{R}$ depends on both, the number of clusters and the number of elements.

Adjusted Rand Index (Hubert & Arabie, 1985) proposed an adjustment of the Rand Index which assumes a generalized hyper-geometric distribution as null hypothesis. In fact, the two clustering are drawn randomly with a fixed number of clusters and a fixed number of elements in each cluster. Thus, the adjusted Rand index represents the normalized difference of the Rand index and its expected value under the null hypothesis. It is given by the following equation:

$$\mathcal{R}_{adj}(C, C') = \frac{\sum_{i=1}^{k}\sum_{j=1}^{l}\binom{m_{ij}}{2} - t_3}{\frac{1}{2}(t_1 + t_2) - t_3} \tag{1.42}$$

where

$$t_1 = \sum_{i=1}^{k}\binom{|C_i|}{2} \tag{1.43}$$

$$t_2 = \sum_{j=1}^{l}\binom{|C'_j|}{2} \tag{1.44}$$

$$t_3 = \frac{2t_1 t_2}{n(n-1)} \tag{1.45}$$

with C_i, C'_j are 2 clusters, m_{ij} is the contingency table of the pair C, C' and n is the cardinality of the set X.

It should be noted that the adjusted Rand index can yield negative values if the index is less than the expected index.

In what follows, we present the networks used in this thesis to evaluate and validate the different proposed approaches based on previous measures.

1.4 Used Network

In this section, we present the used social networks in the experiments of the proposed contributions. In this thesis we used real networks such as the karate club, the dolphins and the books about US politics networks in addition of generated LFR networks. In what follows, the characteristics of each real network such as the number of nodes, links and communities are detailed. As for the generated LFR networks, the list of used parameters is explained.

1.4.1 Karate Club Network

The Zachary Karate Club presented in Figure 1.2 is a well-known social network of an university karate club studied by Zachary (Zachary, 1977). The study was carried out over a period of three years from 1970 to 1972.

In this network, we find:

- 34 nodes that represent the members of Karate Club.
- 78 pairwise links between members who are interacted outside the club.

During the study a conflict arose between the administrator “John A” and instructor “Mr. Hi”, which led to the split of the club into two. Half of the members formed a new club around Mr. Hi, members from the other part found a new instructor or gave up karate.

1.4.2 Dolphins Network

The Dolphins, animals social network presented in Figure 1.3 was introduced by (Lusseau, 2003).

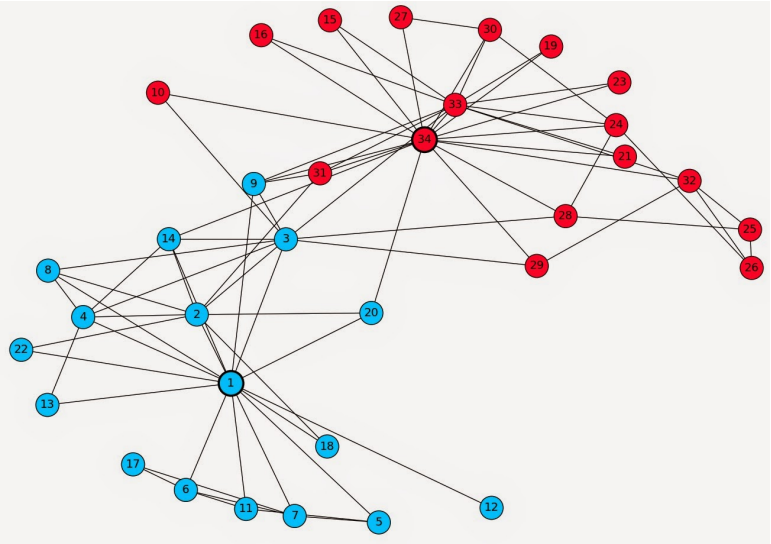

Figure 1.2: The Karate Club Network.

It is composed of 62 bottle-nose dolphins living in Doubtful Sound, New Zealand and social ties established by direct observations over a period of several years. This network is composed of 2 communities and contains 159 edges that indicates a frequent association. The dolphins were observed between 1994 and 2001.

During the course of the study, the dolphins group split into two smaller subgroups following the departure of a key member of the population.

1.4.3 Books about US Politics Network

The network of books [1] presented in Figure 1.4 is composed of 3 communities, 105 nodes and 441 edges that represent books dealing with US politics sold by the on-line bookseller Amazon.com. The edges represent frequent co-purchasing of books by the same buyers.

The books are grouped according to their political spectrum whether they are conservative (represented by red), neural (green) or liberal (blue) based on synopsis and reviews about the books.

[1]The Karate Club, Dolphins and Books about US Politics data sets can be found in http://networkdata.ics.uci.edu/index.php

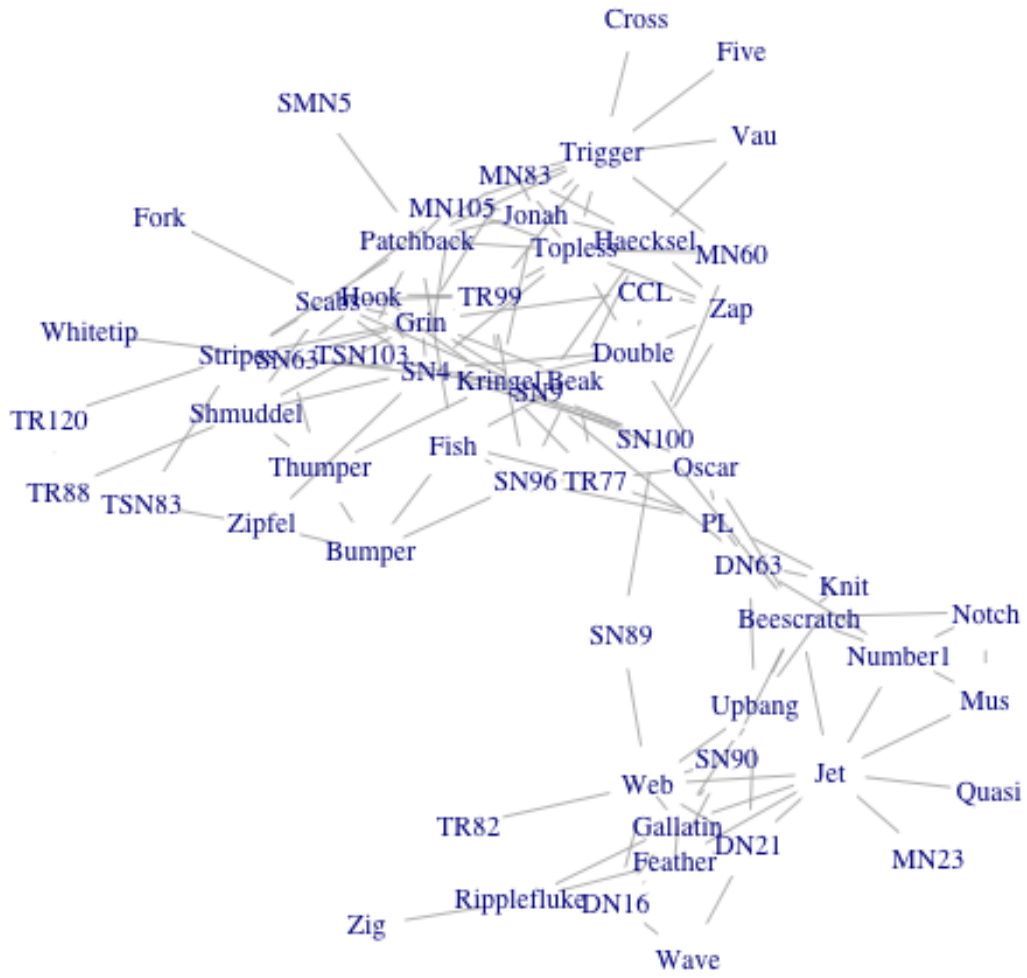

Figure 1.3: Dolphins Network.

1.4.4 LFR Networks

The LFR benchmark [2] (Lancichinetti et al., 2008) is an algorithm that generates artificial networks that simulate real-world networks. The generated network has a prior known communities and it is used to compare different community detection methods. In what follows, we remind the meaning of each parameter of LFR:

- N represents the number of nodes,

[2]The LFR benchmark can be found in https://figshare.com/articles/Lancichinetti-Fortunato-Radicchi-LFR-benchmark/1149962

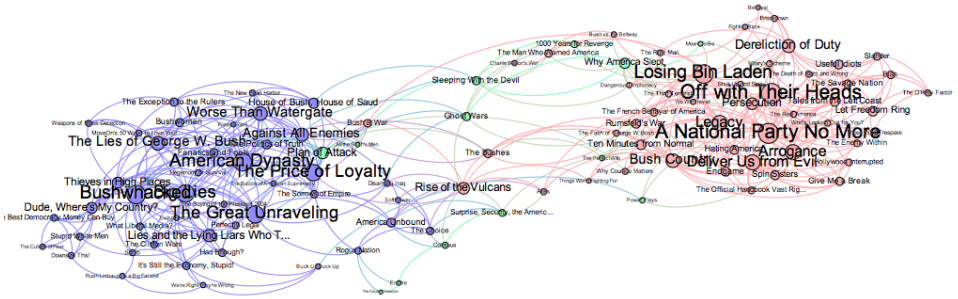

Figure 1.4: Books about US Politics Network.

- k the average degree,
- $maxk$ the maximum degree,
- mu the mixing parameter,
- $t1$ the minus exponent for the degree sequence,
- $t2$ the minus exponent for the community size distribution,
- $minC$ the minimum for the community size,
- $maxC$ the maximum for the community size,
- on the number of overlapping nodes,
- om the number of memberships of the overlapping nodes
- and C the average clustering coefficient.

The LFR benchmark generates networks with power law degree in addition of power law distributed community sizes, and the network size is not constrained.

We present in Appendix A the LFR parameters of the generated networks used in this thesis.

1.5 Conclusion

In this chapter we first introduced the concept of social network as well as its mathematical representation and its analysis. Indeed, we presented the notion of

communities in addition of the social networks properties and some used metrics.

Then, some related works that use uncertainty in this kind of networks were presented. In this thesis, we choose to use the theory of belief functions because it represents a strong tool to deal with imperfect information, to model ignorance and to combine information provided by different sources. Some basic concepts of this theory were presented in this chapter.

After that, different related works to the community detection problems in social networks were introduced. All the cited researches are interesting. However, some works focused only on the structure of the network and neglect the attributes associated to the actors in the social network. Other works take into account both the network structure and the nodes properties but neglect the attributes associated to the links or neglect the fact that the information can be imperfect. Regarding researches that focused on the resolution of the link prediction problem, they are only interested on how to add links to the network when an entity disappears. As for the spammers detection problems, the current works focused on the prediction of the class label of tweets or on analysing the user's profile to predict whether the user is a spammer or not. They do not take into consideration that the link can be spammed. Thereafter, we presented the Walktrap algorithm that will be used in the next chapter in order to compare the obtained classification results of our proposed approach with those of the Walktrap method. Then, some community structure comparison metrics that allow to determine the quality of a classification results were presented.

Finally, we presented the used networks in this thesis as well as their characteristics. These networks allow to evaluate and validate the effectiveness of the proposed approaches.

In the next chapter we present our first contribution which consists on showing the advantage of using evidential attributes in social networks. This method is based on both structure of the network and the attributes associated to the nodes.

To do so, in what follows the obtained NMI results of the clustering of the nodes with different uncertain attributes: numerical, probabilistic and evidential ones are compared.

Chapter 2

The Advantage of Evidential Attributes in Social Networks

2.1 Introduction

Currently, the community detection task becomes important since it allows us to classify the nodes according to their structural position and/or their attributes. Indeed, the clusters obtained by the community detection algorithm contain similar objects.

In this chapter, we present our approach (Ben Dhaou et al., 2017) which consists of detecting communities in the social network using the K-Medoids algorithm. Based on the structure of the network, uncertain attributes are associated to each node. Three types of attributes were used: numerical, probabilistic and evidential. We use the K-Medoids algorithm because of its robustness in the presence of noise and its effectiveness in the case of small data.

The proposed approach is tested on a real data set and some generated LFR networks. The comparison of the clustering quality was evaluated using the Normalized Mutual Information (NMI). In addition, we use other metrics such as the Variation of Information (VI), the Rand index and the adjusted Rand index to compare the obtained clustering results with the ground-truth of each network. The quality of clustering obtained when using a method that only takes into account the structure of the network (walktrap) is also evaluated.

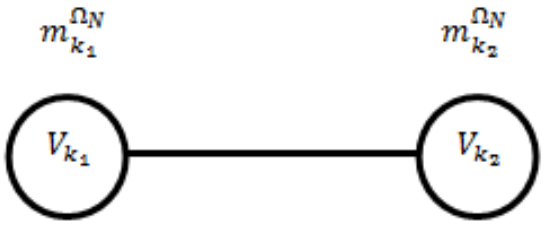

Figure 2.1: Graph with Evidential Attributes on Nodes.

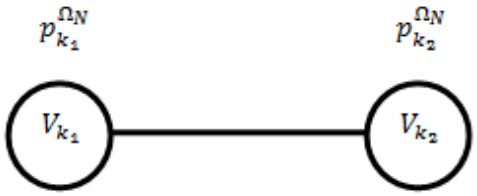

Figure 2.2: Graph with Probabilistic Attributes on Nodes.

This chapter is organized as follows. First, the proposed approach is introduced in section 2.2. Then, the process of experiments and the obtained results are presented in section 2.3. After that, the clustering results comparison with various metrics are shown in section 2.4. Next, in section 2.5 the obtained results in the case of the simple support functions are presented. Section 2.6 details the obtained results given by the Walktrap approach. Finally, section 2.7 concludes the chapter.

2.2 Clustering based on Nodes Attributes

In this contribution, we are interested in the structure of the network as well as the attributes associated with the nodes. Figures 2.1 and 2.2 show respectively a graph with nodes having evidential attributes and a second one with nodes having probabilistic attributes.

In the algorithm 2.1, we propose a method of generating numerical, probabilistic and evidential attributes in order to find communities and show how different attributes make it possible to place each node in its true community.

In the first step, we give a numerical attribute to each node (a single value

$x \in [0, 1]$) which indicates the membership of that node to the community according to the number of communities. We consider the node's class C_i among the set of N possible classes according to the value of x:

$$x \in \left[\frac{i-1}{N}, \frac{i}{N}\right].$$

Two scenarios were considered:

First scenario: We randomly generate the values of the attributes for each node $V_{k_i} \in C_i$ of the graph. We consider three kind of attributes: numerical, probabilistic and evidential.

- Numerical attribute: We generate a value x in $\left[\frac{i-1}{N}, \frac{i}{N}\right]$ for V_{k_i}.
- Probabilistic attribute: We generate a value x in $\left[\frac{i-1}{N}, \frac{i}{N}\right]$ corresponding to the probability $p(V_{k_i} \in C_i)$. For the $n-1$ other probabilities, we generate $N-1$ values in $[0, 1-x]$ that we associate randomly to the other classes. In order to normalize the probability we divide by the sum of the generated values. This process generates n values x_i.
- Evidential attribute: First, we generate a value x in $\left[\frac{i-1}{N}, \frac{i}{N}\right]$ corresponding to the mass function $m(C_i)$. Then the mass of the 2^{N-1} other focal elements containing C_i are generated in $[0, 1-x]$ and randomly associated to the focal elements. At last, we normalize the mass function as in the probabilistic case. This process generates $1 + 2^{N-1}$ values x_i.

Second scenario: In order to avoid the arbitrary level of value on the real class, we affect the highest value to the real class.

- Numerical attribute: In that case, we have only one value, so this second scenario cannot concern the numerical attributes.
- Probabilistic attribute: The maximum of the n values x_i is searched, and the values are swapped.
- Evidential attribute: The maximum of the $1 + 2^{N-1}$ values x_i is searched, and the values are swapped.

After the generation of the attributes of each node, the community detection is made by the K-Medoids algorithm which is robust in the presence of noise. Moreover, this algorithm is interesting and effective in the case of small data. In the case of evidential attributes, we use the distance of Jousselme (Jousselme et al., 2001) between the attributes.

After that, we compare the obtained clusters with the real clusters. In order to measure the clustering quality in each cluster, we use the Normalized Mutual Information (NMI), a measure that allows a compromise between the number of clusters and their quality (Knops et al., 2006).

In a second step, in order to evaluate the robustness of the proposed approach, we select randomly few nodes of the graph and modify their classes. Then, we compute again the NMI and compute the Interval of Confidence.

Algorithm 2.1 shows the outline of the process followed for evidential attributes before adding the noisy attributes in both first and second scenarios. Algorithm 2.2 presents the process used with the evidential attributes after adding the noisy nodes in both scenarios.

Algorithm 2.1 Generation of Evidential Attributes

Require: G: Network,
n: Number of vertices,
K: Number of clusters,
C_i: Elements of each cluster i
Ensure: *nmiAttr*: Similarities between evidential attributes, *IC*: Interval of Confidence
// First Scenario: Random Generation
for all $V_{k_i} \in C_i$ **do**
EvidentialLabels(C_i)
// a function that generates randomly mass functions according to some conditions for each node belonging to C_i.
//Second Scenario
Sort(EvidentialLabels)
// Put the highest generated value on the attribute "C_i" according to which community, the node belongs and the rest on the subsets containing "C_i".
end for

Algorithm 2.2 Adding Noisy Attributes

Require: G: Network,
n: Number of vertices,
K: Number of clusters,
C_i: Elements of each cluster i
V_{k_i}: Labelled vertices

Ensure: *nmiAttr*: Similarities between evidential attributes, *IC*: Interval of Confidence

// Adding Noisy Attributes in both scenarios.
Select randomly n nodes of the network and modify their attributes in order to modify their classes.

2.3 Experimentations

In this section we perform some experiments on real networks from the UCI data sets, such as the Karate Club network, the Dolphins network and the Books about US Politics network in addition of few LFR Networks.

Process of Experimentations

The purpose of these experiments is to compare the obtained clustering results with the different uncertain attributes before and after adding noisy attributes. In these experiments, the attributes are first generated based on the structure of each network:

Numerical Attributes: For this type of attribute, a single value is generated.

1. Karate Club: This network has 2 communities, so a single value of attribute is given to each node belonging to C_1 in the interval $[0, 0.5]$ and a value in $[0.5, 1]$ if the node belongs to C_2.

2. Dolphins Network: This network has also 2 communities. The same intervals as those of the Karate Club are chosen: if the node belongs to C_1, we generate an attribute in $[0, 0.5]$ and in $[0.5, 1]$ if the node belongs to C_2.

3. Books about US Politics Network: This network has 3 communities: For the node belonging to C_1, an attribute in $[0, 0.33]$ is given. Each node belonging

to C_2 has an attribute in $[0.33, 0.66]$. Finally, for the nodes of C_3, they have an attribute in $[0.66, 1]$.

4. LFR Networks: All the generated networks have 3 communities. The same intervals as those of the Books about US Politics Network are used: if the node belongs to C_1, we associate an attribute in $[0, 0.33]$. For each node belonging to C_2, an attribute in $[0.33, 0.66]$ is given. At last, an attribute in $[0.66, 1]$ is assigned to the nodes belonging to C_3.

Probabilistic Attributes: For this type of attributes, 2 or 3 values are generated depending on the type of network.

1. Karate Club: For the nodes belonging to C_1, they have a first value picked randomly in the interval $[0, 0.5]$ and the second value is deduced from that $(1 - x)$. For the elements of C_2, the first values of attributes were picked randomly from the interval $[0.5, 1]$ and the second ones are deduced from that $(1 - x)$.

2. Dolphins Network: Same thing as for the karate club, the nodes of C_1 have a first value of attribute in $[0, 0.5]$ and the second value is deduced from that. The nodes of C_2 have a first value in $[0.5, 1]$ and the second one is deduced from the first one.

3. Books about US Politics Network: For the nodes of C_1, their first value of attributes is picked in the interval $[0, 0.33]$, the second and third values are generated randomly from $[0, (1 - x)]$. After that, we normalize by dividing the second and the third probabilities by the sum of the first, second and third probabilities. For the nodes of C_2 and C_3 the same process is followed, except that the first values of the attributes are picked from the interval $[0.33, 0.66]$ in the case of the elements belonging to C_2 and from the interval $[0.66, 1]$ for the nodes of C_3.

4. LFR Networks: For all the generated networks, the first value of the attributes of the nodes belonging to C_1 is picked in the interval $[0, 0.33]$ and for the rest of the values, they are generated randomly from $[0, (1 - x)]$. We normalize by dividing the second and third probabilities by the sum of the first, second and third probabilities. Regarding the nodes belonging to C_2, the same process is followed but instead of picking the first value in $[0, 0.33]$,

we use the interval $[0.33, 0.66]$. Finally, for the nodes of C_3, the first value is picked in $[0.66, 1]$.

Evidential Attributes: For this type of attributes 2 and 4 values are generated, depending on the type of network.

1. Karate Club: This network has 2 communities so, $\Omega_N = \{C_1, C_2\}$ and $2^{\Omega_N} = \{\emptyset, C_1, C_2, C_1 \cup C_2\}$. We choose to put 2 values on C_1 and Ω_N for the nodes belonging to C_1. For the rest of hypothesis, we put 0. For the value of C_1, it was picked in the interval $[0, 0.5]$ and the second value on Ω_N was deduced from the first value. We remind that the sum should be equal to 1. For the nodes of C_2, 2 values are affected to C_2 and Ω_N. The first value of C_2 is picked in $[0.5, 1]$ and the second one is deduced of the first value.

2. Dolphins Network: The same process used for the Karate Club is applied.

3. Books about US Politics: this network has 3 communities so, the frame of discernment $\Omega_N = \{C_1, C_2, C_3\}$ and the power set $2^{\Omega_N} = \{\emptyset, C_1, C_2, C_1 \cup C_2, C_3, C_1 \cup C_3, C_2 \cup C_3, C_1 \cup C_2 \cup C_3\}$. Four values are affected to C_1, $C_1 \cup C_2$, $C_1 \cup C_3$ and Ω_N when the nodes belong to C_1. For the rest of the hypothesis, the value 0 is assigned. The value of C_1 is picked from $[0, 0.33]$ and the rest of the values are deduced from the first one. The same principle as deducing the rest of probabilities presented previously is used, except that 3 other probabilities are generated instead of 2. For the second community, the same process is applied, except that we put values on C_2 and the subsets containing C_2. The value of C_2 is picked in the interval $[0.33, 0.66]$ and the rest of the values were deduced as explained before. For the third community, the values are generated on C_3, and each subset containing C_3. The value of C_3 is picked in $[0.66, 1]$ and the rest of the values are deduced as explained before.

4. LFR Networks: As all generated networks are composed of 3 communities, same process used for the Books about US Politics Network is applied.

Once the attributes generated, the K-medoids algorithm is used to cluster the nodes according to their attributes. After that, the NMI method is used to compare the detected clusters with the real clusters of each network. Then, we compute the confidence interval. These experimentations are repeated 100 times.

	NMI-Average	Interval of Confidence
Numerical	0.776	[0.596, 0.955]
Probabilistic	0.778	[0.59, 0.967]
Evidential	1	[1, 1]

Table 2.1: NMI Averages et Intervals of Confidence- Case of Karate Club: First Scenario.

In a second time, the generated matrices are sorted by putting the highest values on C_1 and C_2 in the case of the Karate Club and Dolphins network and on C_1, C_2 and C_3 in the case of the Books about US Politics network and the LFR networks. After that, the nodes are clustered again according to their new attributes and the NMI averages are computed.

The second part of the experimentations consists on adding some noisy attributes by modifying the attributes of some nodes of C_1, C_2 and C_3. For each noisy attribute, its value is chosen outside the interval set for its class. Then, the nodes are clustered according to their attributes and the NMI average as well as the intervals of confidence are computed. This experimentation is performed for the random and the sorted matrix of attributes. It should be noted that the sorted attributes matrices are used in the case of the probabilistic and the evidential generation only. In the results below, we present the average of NMI computed for 100 executions of the algorithms and the intervals of confidence for the numerical, probabilistic and evidential attributes.

2.3.1 Results Before Adding the Noise

Karate Club Network

First Scenario In this section, we show the results of the NMI computation of the random generated attributes. The results of the average values of NMI for 100 runs of random attributes generation are presented below.

Table 2.1 shows that the evidential generated attributes give better results than the probabilistic and the numerical ones. In fact, a value of the NMI average equal

	NMI-Average	Interval of Confidence
Probabilistic	0.7843	[0.602, 0.966]
Evidential	1	[1, 1]

Table 2.2: NMI Averages et Intervals of Confidence- Case of Karate Club: Second Scenario.

	NMI-Average	Interval of Confidence
Numerical	0.782	[0.587, 0.976]
Probabilistic	0.765	[0.554, 0.976]
Evidential	1	[1, 1]

Table 2.3: NMI Averages et Intervals of Confidence- Case of Dolphins Network: First Scenario.

to 1 is obtained which means that the K-medoids is able to classify the nodes according to their evidential attributes in the right cluster even when the generation is random.

Second Scenario The generation of the attributes is executed several time and the matrix of attributes is sorted (We put the highest value on the attribute C_1 or C_2 depending on the belonging of the node to C_1 or C_2). The obtained results of the average values of NMI for 100 executions are shown below.

The results presented in Table 2.2 show that the evidential version gives an average NMI value equal to 1, which means that each node was detected in the right cluster. It is noticed that after sorting the probabilistic attributes, the K-medoids was not able to affect all the nodes in their right cluster.

Dolphins Network

First Scenario The average values of NMI for 100 runs of random generated attributes in the Dolphins network are presented in Table 2.3.

	NMI-Average	Interval of Confidence
Probabilistic	0.79	[0.597, 0.983]
Evidential	1	[1, 1]

Table 2.4: NMI Averages et Intervals of Confidence- Case of Dolphins Network: Second Scenario.

	NMI-Average	Interval of Confidence
Numerical	0.699	[0.551, 0.848]
Probabilistic	0.758	[0.668, 0.848]
Evidential	1	[1, 1]

Table 2.5: NMI Averages et Intervals of Confidence- Books about US Politics Network: First Scenario.

The obtained average NMI in the case of evidential attributes is the highest value comparing to the probabilistic and the numerical ones. Same thing, the K-medoids is able to classify the nodes in their right cluster based on their evidential attributes.

Second Scenario The matrix of the previous generated attributes is sorted and the average values of NMI are computed for 100 executions.

The results presented in Table 2.4 show that the evidential version gives an average NMI value equal to 1 comparing to the probabilistic and numerical versions. It is also noticed that the K-medoids was not able to classify the nodes in their right clusters based on their probabilistic attributes. In fact, only 79% of nodes were classified correctly compared to 100% corrects nodes having evidential attributes.

Books about US Politics Network

First Scenario In this part, the obtained results of the NMI average values are shown in the case of 100 runs of random generated attributes.

	NMI-Average	Interval of Confidence
Probabilistic	0.895	[0.828, 0.962]
Evidential	1	[1, 1]

Table 2.6: NMI Averages et Intervals of Confidence- Books about US Politics Network: Second Scenario.

The results in Table 2.5 show that the clustering based on the generated evidential attributes gives better results than the probabilistic and the numerical ones. In fact, the evidential NMI average is equal to one which means that all the nodes were classified in their right cluster.

Second Scenario The generation of the attributes is performed several times and the matrix of attributes is sorted (the highest value is assigned to the attribute C_1, C_2 or C_3 depending of the belonging of the node to the first, second or third community). The results of the average values of NMI for 100 are presented below.

The obtained results in Table 2.6 show that the evidential version gives an average NMI value equal to 1 comparing to the probabilistic one which means that all the nodes were classified in their right clusters.

LFR Network: 300 Nodes + 3 Communities

First Scenario In this section, we show the results of the NMI computation of the clustering results of the random generated attributes. Table 2.7 presents the obtained average values of NMI for 100 runs of random attributes generation in the case of an LFR network composed of 300 nodes and 3 communities.

The results show that the evidential generated attributes give better results than the probabilistic and the numerical ones. In fact, we obtained a value of the NMI average equal to 1 which means that the K-medoids succeeded to classify all the nodes, according to their evidential attributes, in their right cluster.

	NMI-Average	Interval of Confidence
Numerical	0.659	[0.615, 0.703]
Probabilistic	0.676	[0.627, 0.725]
Evidential	1	[1, 1]

Table 2.7: NMI Averages et Intervals of Confidence- LFR 300 Nodes: First Scenario.

	NMI-Average	Interval of Confidence
Probabilistic	0.856	[0.833, 0.879]
Evidential	1	[1, 1]

Table 2.8: NMI Averages et Intervals of Confidence- LFR 300 Nodes: Second Scenario.

Second Scenario The generation of the attributes is executed several time and we sorted the matrix of attributes. The obtained results of the average values of NMI for 100 executions are presented below.

The results of Table 2.8 show that the evidential version gives an average NMI value equal to 1, which means that each node was affected to the right cluster. It is noticed that after sorting the probabilistic attributes, the K-medoids was able to affect only 85% of the nodes in their right clusters.

It has been noticed that either in the case of the first scenario or the second one, the NMI values obtained with evidential attributes are always equal to 1. This can be explained by the fact that a mass function is assigned to each hypothesis containing the class to which the node belongs. The performed experiments on other different LFR networks are presented in Appendix B .

2.3.2 Results After Adding the Noise

In this section, the obtained results after adding some noisy attributes are presented. To do so, 1 to 9 nodes of the real data networks are randomly chosen on which some noise is added. Regarding the LFR generated networks, the noise is added according to the networks sizes. For the LFR network composed of:

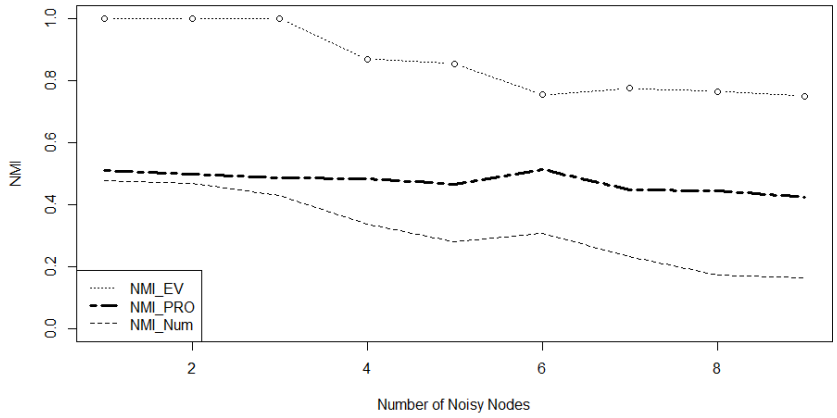

Figure 2.3: Noisy Karate: First Scenario.

- 50 nodes, we added from 5 to 25 noisy nodes.
- 99 nodes, we added from 5 to 25 noisy nodes.
- 200 nodes, we added from 10 to 50 noisy nodes.
- 300 nodes, we added from 20 to 60 noisy nodes.

Hence, the attributes values are modified and the NMI average values are computed each time. This experimentation is repeated 100 times for each number of modified nodes, for cross-validation.

First Scenario At first, the first scenario is considered and the results obtained on Karate Club dataset are presented in figure 2.3, on Dolphins dataset in figure 2.4, on Books about US Politics dataset in figure 2.5 and on LFR Network composed of 50, 99, 200 and 300 nodes respectively in figures 2.6, 2.7, 2.8 and 2.9.

From the different curves of the real data, it is deduced that the evidential attributes allow the K-medoids to cluster the nodes in their right clusters better than the numerical and the probabilistic attributes. In fact, it is noticed that with the evidential attributes, almost all the nodes are classified in their right clusters even when the number of the noisy nodes is equal to 9. In addition, the intervals of confidence show that the evidential attributes are better than the probabilistic

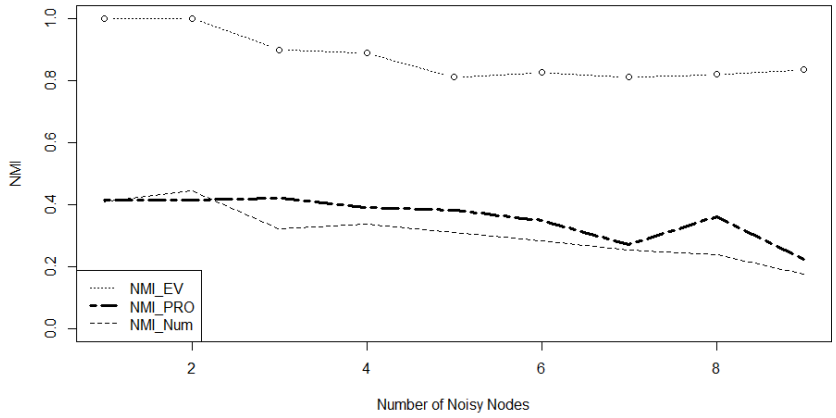

Figure 2.4: Noisy Dolphins: First Scenario.

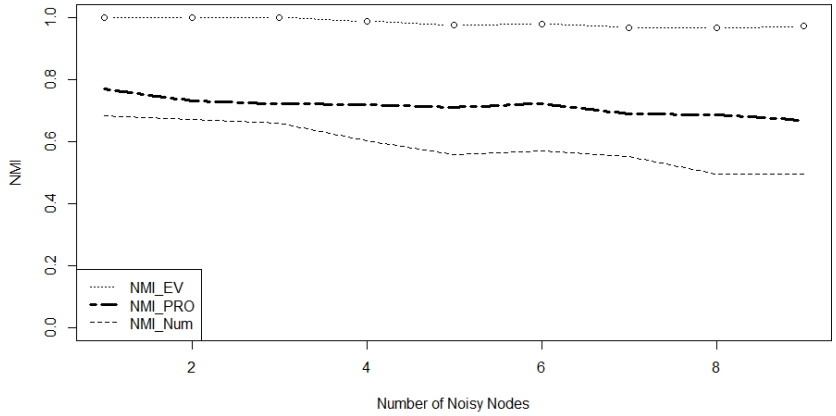

Figure 2.5: Noisy Books: First Scenario.

and numerical ones. For example, for 3 noisy nodes, the interval of confidence in the case of the karate club network is equal to: $[0.329, 0.531]$ for the numerical version, $[0.309, 0.63]$ for the probabilistic version and $[1, 1]$ for the evidential version.

In the case of the Dolphins network, the interval of confidence is equal to: $[0.258, 0.389]$ for the numerical version, $[0.301, 0.541]$ for the probabilistic version and $[0.864, 0.935]$ for the evidential version.

Moreover, in the case of the Books about US Politics network, the interval of

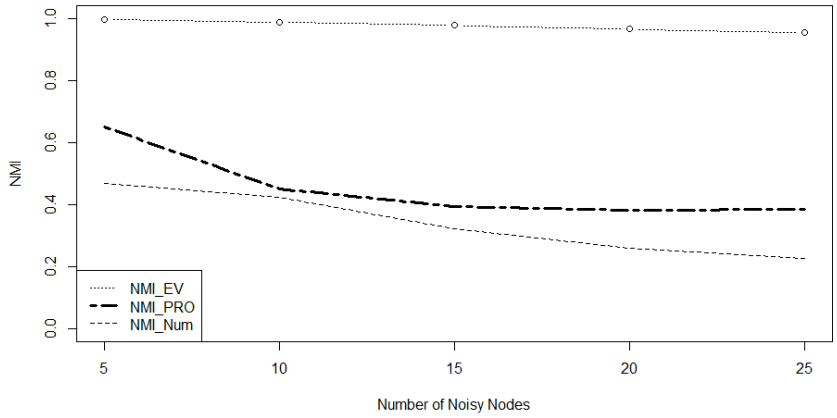

Figure 2.6: Noisy LFR 50N: First Scenario.

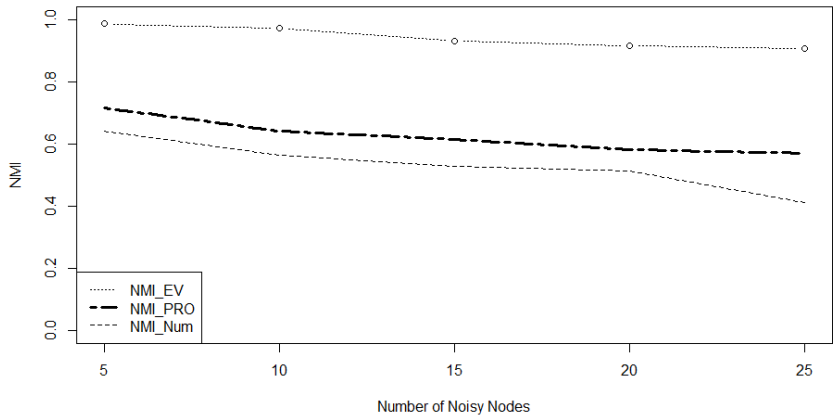

Figure 2.7: Noisy LFR 99N: First Scenario.

confidence is equal to: $[0.554, 0.762]$ for the numerical version, $[0.649, 0.795]$ for the probabilistic version and $[1, 1]$ for the evidential version.

Figure 2.6 shows the NMI average values given by the numerical, probabilistic and evidential attributes. It is noticed that the K-Medoids gives better clustering results when the evidential attributes are used. The noise was varied from 5 to 25 noisy nodes. Regarding the confidence intervals, they confirm the obtained results. Indeed, if the case of 15 noisy nodes is considered, we have $[0.269, 0.374]$ in the case of the numerical attributes, $[0.363, 0.424]$ for the probabilistic ones and

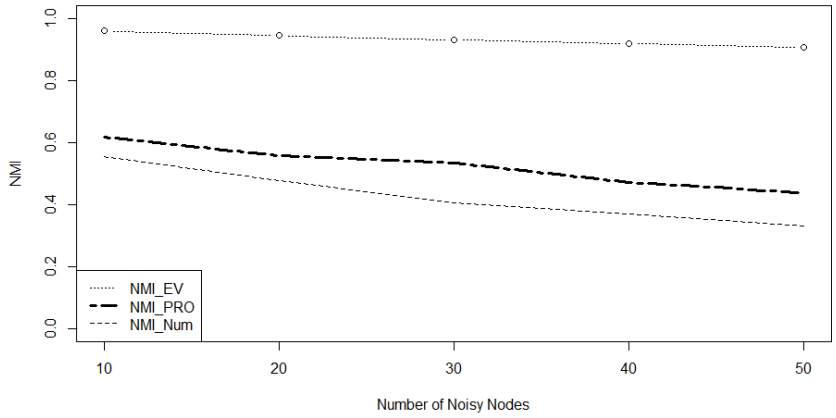

Figure 2.8: Noisy LFR 200N: First Scenario.

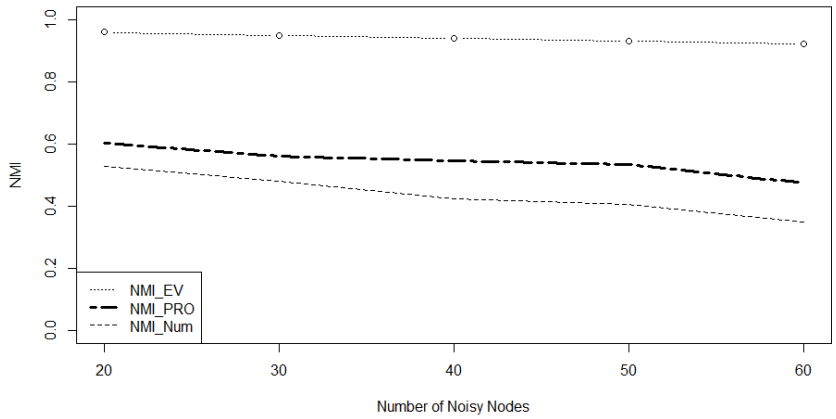

Figure 2.9: Noisy LFR 300N: First Scenario.

$[0.889, 1]$ in the case of evidential attributes.

Figure 2.7 represents also the NMI values obtained from numerical, probabilistic and evidential attributes. It is remarked that the clustering based on the evidential ones gives better results. The noise is varied from 5 to 25 noisy nodes. The intervals of confidence confirm the previous results. Indeed, if the case of 25 nodes is considered, we have $[0.364, 0.459]$ in the case of numerical attributes, $[0.471, 0.668]$ for the probabilistic ones and $[0.865, 0.923]$ for the evidential attributes.

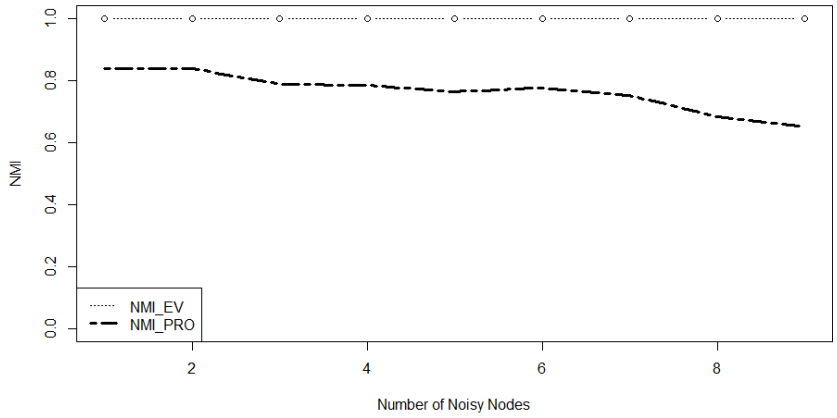

Figure 2.10: Noisy karate: Second Scenario.

From the curve of the LFR Network composed of 200 nodes, it is noticed that the evidential attributes allow the K-medoids to affect the nodes to their right clusters. The noise is varied from 10 to 50 noisy nodes. When the intervals of confidence are considered in the case of 30 noisy nodes, we have $[0.35, 0.46]$ for the numerical attributes $[0.502, 0.565]$ in the case of the probabilistic attributes and $[0.929, 0.945]$ for the evidential ones.

Figure 2.9 shows the NMI values obtained from numerical, probabilistic and evidential attributes. The noise is varied from 20 to 60 noisy nodes. The curve shows that a better clustering results is obtained with evidential attributes. Regarding the intervals of confidence, we have $[0.406, 0.441]$ in the case of numerical attributes, $[0.519, 0.57]$ in the case of probabilistic ones and $[0.925, 0.949]$ in the case of evidential attributes when the case of 40 noisy nodes is considered.

Second Scenario Now, the second scenario is considered and the results obtained on Karate Club dataset are presented in figure 2.10, on Dolphins dataset in figure 2.11, on Books about US Politics dataset in figure 2.12 and on LFR Network respectively in figure 2.13 for the LFR network composed of 50 nodes, figure 2.14 for the LFR composed of 99 nodes, figure 2.15 for the LFR composed of 200 nodes and finally figure 2.16 for the LFR composed of 300 nodes.

The results show that the clustering based on the evidential attributes gives better results than the probabilistic attributes. Indeed, the nodes with the evidential

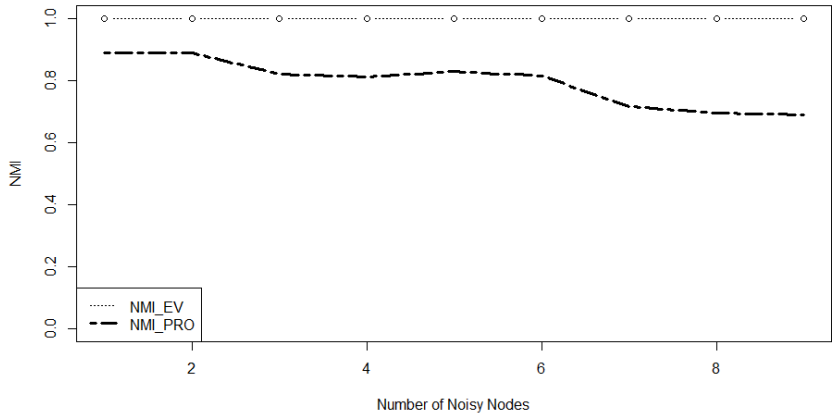

Figure 2.11: Noisy Dolphins: Second Scenario.

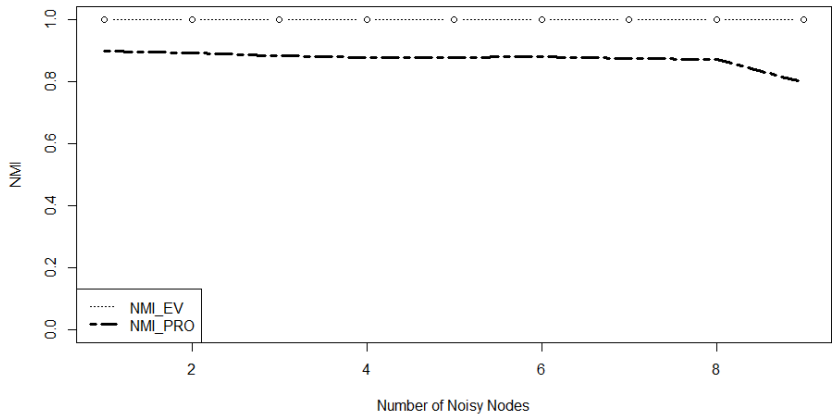

Figure 2.12: Noisy Books: Second Scenario.

attributes are almost all classified in their right clusters. In addition, the intervals of confidence show that the evidential attributes are better than the probabilistic ones. For example, for 3 noisy nodes, the interval of confidence in the case of the karate club network is equal to: $[0.718, 0.856]$ for the probabilistic version and $[1, 1]$ for the evidential version.

In the case of the Dolphins network, the interval of confidence is equal to: $[0.794, 0.843]$ for the probabilistic version and $[1, 1]$ for the evidential version.

In the case of the Books about US Politics network, the interval of confidence

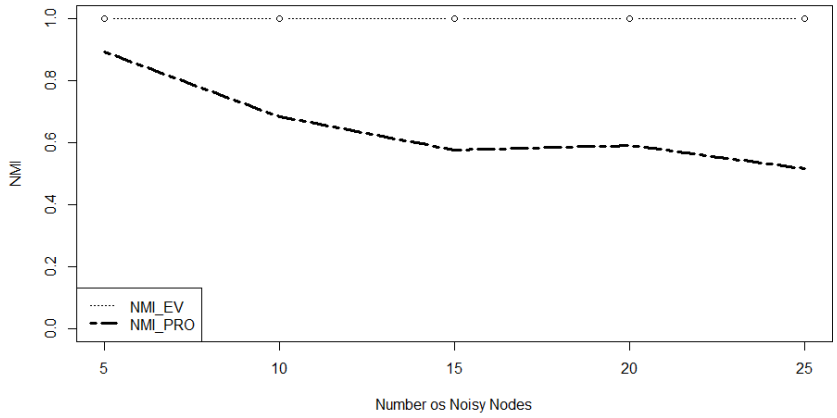

Figure 2.13: Noisy LFR 50N: Second Scenario.

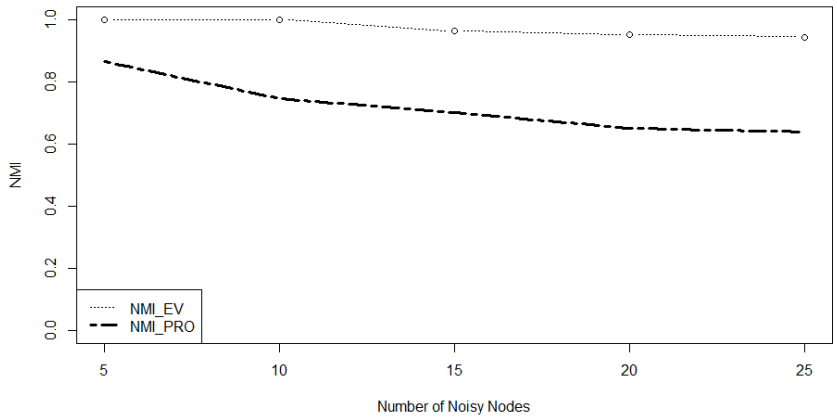

Figure 2.14: Noisy LFR 99N: Second Scenario.

is equal to: $[0.836, 0.933]$ for the probabilistic version and $[1, 1]$ for the evidential version.

Figure 2.13 shows the obtained NMI averages with the numerical, probabilistic and evidential attributes. The noise was varied from 5 to 25 noisy nodes. It is also noticed that the intervals of confidence confirm that we obtain better clustering results with the evidential attributes. If the case of 15 noisy nodes is considered, we have $[0.517, 0.633]$ in the case of probabilistic attributes and $[1, 1]$ with the evidential ones. Therefore, all the nodes with evidential attributes were

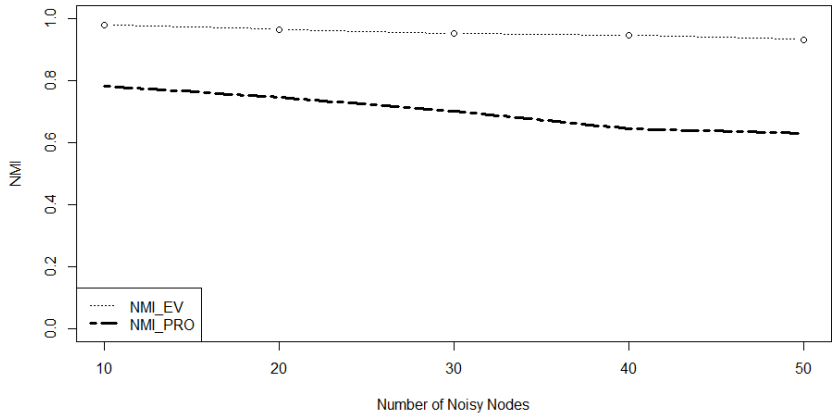

Figure 2.15: Noisy LFR 200N: Second Scenario.

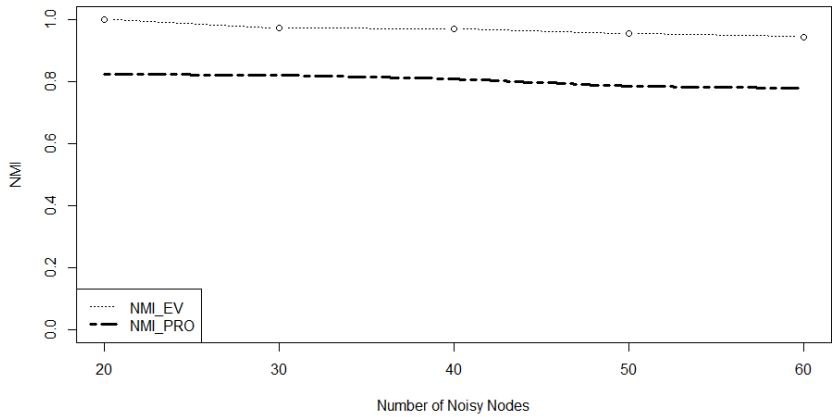

Figure 2.16: Noisy LFR 300N: Second Scenario.

all classified in their right clusters.

Regarding the LFR Network composed of 99 nodes, the curves show that the clustering with the evidential attributes gives better results than the other type of attributes. The noise is varied from 5 to 25 noisy nodes. The intervals of confidence show that the evidential attributes are better than the probabilistic ones. Indeed, in the case of the probabilistic attributes, we have $[0.61, 0.665]$ while we have $[0.925, 0.953]$ for the evidential ones in the case of 25 noisy nodes.

The curve of the LFR Network composed of 200 nodes shows that the eviden-

tial attributes allow the K-medoids to place the nodes in their right clusters. The noise is varied from 10 to 50 noisy nodes. When the intervals of confidence are considered in the case of 30 noisy nodes, we have $[0.664, 0.736]$ in the case of the probabilistic attributes and $[0.93, 0.966]$ for the evidential ones.

The curve of the LFR Network composed of 300 nodes shows that better clustering results are obtained with evidential attributes. The noise was varied from 20 to 60 noisy nodes. Regarding the intervals of confidence, we obtain $[0.791, 0.827]$ in the case of probabilistic attributes and $[0.952, 0.978]$ in the case of evidential attributes when we add 40 noisy nodes.

Whether the obtained results with real networks or generated ones after adding noise, the clustering with the evidential attributes gives the best NMI results. This is because the theory of belief functions manages better than other frameworks the ignorance and uncertainty.

In order to confirm the effectiveness and the advantage of using the evidential attributes in the clustering, the quality of the clustering is tested with other various metrics in the next section.

2.4 Clustering Results Comparison with various Metrics

We present in this section a comparison of the clustering results with various metrics in the case of adding 6 noisy nodes in Dolphins Network and 15 nodes in LFR Network. The presented results are the average of 100 runs.

We remind in following the meaning of the used metrics:

- *NMI* (Knops et al., 2006): It is a good measure for determining the quality of clustering. The NMI has a value between 0 and 1 with 0 indicates that there is no mutual information and 1 indicates that it's a perfect correlation.
- *VI* (Meilă, 2003): The variation of information measures the amount of information lost and gained in changing from clustering C to clustering C'. It measures rather difference than similarity, its values are not between 0 and 1. Indeed, it is something between 0 and $2logK$, with K is the number of clusters.

Attributes	NMI-Average	VI-Average	Rand-Average	Adjusted Rand-Average
Numerical	0.284	0.948	0.637	0.275
Probabilistic	0.349	0.863	0.668	0.336
Evidential	0.826	0.217	0.942	0.884

Table 2.9: Community Structures Comparison using various Metrics: Case of Dolphins Network-First Scenario

- *Rand Index* (Rand, 1971) : It is a measure of the similarity between 2 data clusterings. The Rand Index has a value between 0 and 1, with 0 indicating that the two data clusterings do not agree on any pair of points and 1 indicating that the data clusterings are exactly the same.

- *Adjusted Rand Index* (Hubert & Arabie, 1985): It rescales the index taking into account that random chance will cause some objects to occupy the same clusters. The Adjusted Rand Index can yield negative values if the index is less than the expected index.

2.4.1 Dolphins Network

The quality of clustering with the different uncertain attributes is tested in the case of real network "Dolphins" in both first and second scenarios.

First Scenario Table 2.9 shows that the highest clustering quality is obtained when the evidential attributes are used.

Second Scenario In the second scenario, we compare the obtained results with the K-medoids in the case of probabilistic and evidential attributes. It is noticed that all the metrics in table 2.10 confirm that all the nodes were affected to their right clusters.

Attributes	NMI-Average	VI-Average	Rand-Average	Adjusted Rand-Average
Probabilistic	0.813	0.233	0.936	0.871
Evidential	1	0	1	1

Table 2.10: Community Structures Comparison using various Metrics: Case of Dolphins Network-Second Scenario

Attributes	NMI-Average	VI-Average	Rand-Average	Adjusted Rand-Average
Numerical	0.322	1.46	0.671	0.259
Probabilistic	0.393	1.31	0.726	0.383
Evidential	0.977	0.05	0.989	0.976

Table 2.11: Community Structures Comparison using various Metrics: Case of LFR 50N-First Scenario

2.4.2 LFR Network

After making tests in the case of a real network, the obtained results in the case of a generated network are presented in the following.

First Scenario In this part, we compare the quality of the obtained clustering results in the case of random generation of the numerical, probabilistic and evidential attributes with various metrics. Table 2.11 shows that all the metrics confirm that the clustering with the evidential attributes is better than the numerical and probabilistic ones.

Second Scenario In table 2.12, we can notice that when the highest generated values are affected to the communities depending on the belonging of the nodes, the K-medoids succeeds to cluster all the nodes having evidential attributes.

In order to test the proposed approach on larger networks with more communities, we use simple support functions during generation. The obtained results are presented in what follows.

Attributes	NMI-Average	VI-Average	Rand-Average	Adjusted Rand-Average
Probabilistic	0.575	0.932	0.827	0.603
Evidential	1	0	1	1

Table 2.12: Community Structures Comparison using various Metrics: Case of LFR 50N-Second Scenario

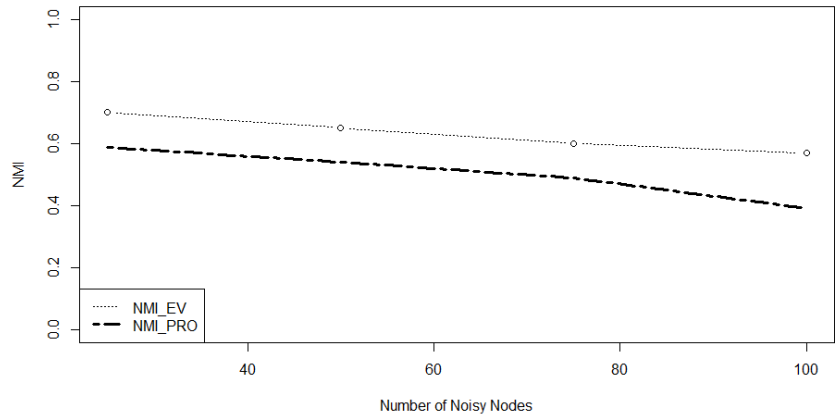

Figure 2.17: Noisy LFR 4 Communities: Second Scenario.

2.5 Simple Support Functions Results

In this section, we present the obtained NMI results in the case of generating simple support functions associated to the nodes based on the structure of the network. A simple support function is a mass function composed of two focal elements, one can be everywhere and the second one should be on Ω_N. In our case, the first generated focal element is on the class C_i to which the node belongs and it takes the highest generated value.

The evidential and probabilistic attributes are compared after adding noise on 3 LFR networks composed of 200 nodes each and have respectively 4, 5 and 6 communities. The noise consists on generating randomly a simple support functions and randomly a vector of probabilities. In Figures 2.17, 2.18 and 2.19, the noise is added to 25, 50, 75 and 100 nodes.

The curves in Figures 2.17, 2.18 and 2.19 show that the more the noise in-

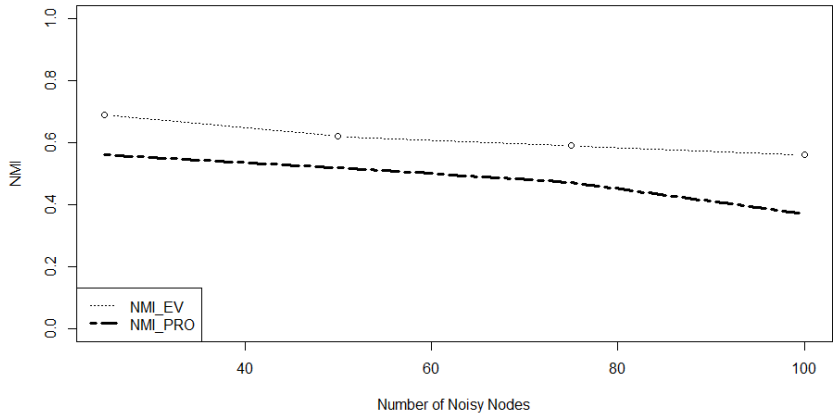

Figure 2.18: Noisy LFR 5 Communities: Second Scenario.

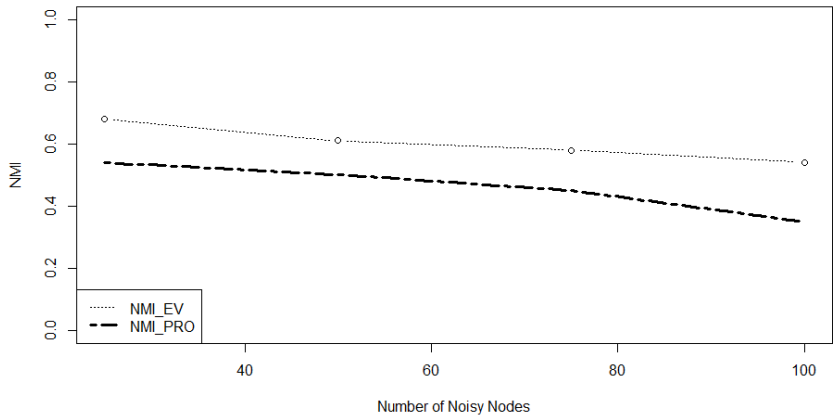

Figure 2.19: Noisy LFR 6 Communities: Second Scenario.

creases, the more the values of the NMI decrease. However, it can be noticed that the nodes with the evidential attributes are better classified than those with the probabilistic ones. This is because the theory of belief functions offers a strong mathematical tool for the management of ignorance and uncertainty whereas in the case of probability theory, ignorance is handled by equi-probabilities.

In order to compare the effectiveness of the proposed algorithm, the obtained results are compared with those given by Walktrap approach which only takes into consideration the structure of the network.

Networks	NMI-Walktrap
Karate Club	0.695
Dolphins Network	0.106
Books about US Politics	0.529
LFR 50N	0.223
LFR 99N	0.032
LFR 200N	0.052
LFR 300N	0.049

Table 2.13: NMI Results given by the Walktrap Method.

2.6 Results given by the Walktrap Approach

The Walktrap approach (Pons & Latapy, 2005), presented in the first chapter, uses a distance measure based on random walks and applies a hierarchical agglomerative clustering algorithm. A random walker is an agent moving from one node to another following the network edges. At each time step, the next node is selected by randomly picking a neighbour of the current node. The idea behind this algorithm is that random walks tend to get trapped into a community. If two nodes i and j are in the same community, the probability to get to a third node k located in the same community through a random walk should not be very different for i and j . The distance is constructed by summing these differences over all nodes, with a correction for degree.

Table 2.13 shows the results of the obtained NMI after comparing the clustering results given by the walktrap method with the actual clusters. It is noticed that the method is not good when we apply it with the networks generated by LFR as well as with the Dolphins network. With regard to the rest of the networks, we find that this method correctly detects only 52% of the nodes of the network Books about US Politics and 69% of the nodes of the network Karate Club.

It can be concluded that considering both structure of the network and attributes of the nodes leads to obtain better NMI results than considering only the network structure.

2.7 Conclusion

In this chapter, different type of attributes (numerical, probabilistic and evidential) were generated and compared in order to determine which one permit to obtain better clustering results.

To do so, two types of experiments are performed: First, clustering the nodes using their attributes generated according to the structure of the networks. In this part, two scenarios were observed: the first one consists on generating randomly the attributes and then performing the K-medoids algorithms to cluster the vertices. The second one consists on putting the highest generated values on the attributes corresponding to the class of each node.

The second type of experiments consists on adding and varying noise to the network and then performing the clustering. Two scenarios were also tested: in the first one, we took the previous random generation and selected some nodes in order to modify their classes. The second scenario consists on considering the case where the highest values were affected to the corresponding classes of the nodes then adding some noisy vertices.

The algorithms were applied on real data set such as the Karate Club Network, the Dolphins network and the Books about US Politics. In addition, the proposed approach was tested on some generated LFR networks.

The obtained results show that in all the scenarios, better clustering quality is obtained with the evidential attributes. This is due to the fact that the theory of belief functions manages better the uncertainty, ignorance and imprecision than the other uncertain theories.

In this chapter it has been shown that, by using the network structure and assigning evidential attributes to the nodes, we have obtained the best clustering results compared to probabilistic and numerical attributes.

In the following chapter, uncertain networks whose nodes and links have attributes that have been associated based on the structure of the network in order to correct the noise added to the information of the nodes and links composing the network are considered.

Chapter 3

An Evidential Method for Correcting Noisy Information

3.1 Introduction

In this chapter, a method (Ben Dhaou et al., 2018) which allows the classification based on the structure of the network as well as on the attributes of the nodes and the links is introduced. The purpose of the proposed method is to correct noisy information in the network and to ensure a coherent network even in the presence of a large amount of noisy information.

In order to evaluate the robustness of the proposed approach, some noise is added by modifying the initial attributes of the nodes as well as the links. The algorithm is tested in 3 scenarios: first, only the vertices attributes are modified, then, we modify the links attributes and finally, the nodes and links attributes are simultaneously modified.

The proposed approach is tested on real data set: the Karate club network and generated LFR networks. The obtained results are compared with those of the probabilistic version of the algorithm.

This chapter is structured as follows. Section 3.2 details the steps of the proposed algorithm. In section 3.3, the process of experiments and all the obtained results are presented. Finally, section 3.4 concludes the chapter.

3.2 Noisy Information Correction based on Nodes and Links Attributes

In this section, the proposed approach is introduced. First, the important notions used in this contribution are presented. Then, the formalization of our method is explained and finally, the main steps of the proposed algorithm are explained.

3.2.1 Noise and Consistency

In the networks, noisy or imperfect information can transit. Therefore, if we limit ourselves to the network structure as well as the nodes and links attributes in the classification, the error rate may increase and the network information may become inconsistent.

To solve this problem, we propose a method that allows the classification of nodes in the case of a noisy network, based on the community structure as well as the nodes and links attributes.

In the case of a significant noise introduced, the algorithm corrects inconsistent information. Thus, even if the initial network is not found, a new coherent network is obtained. In this context, two notions used in this work are presented:

Noise A noisy element (*i.e.* a node or a link) is an element whose attribute has been modified.

Consistency A network is composed of a set of nodes belonging to communities C_i and linked together by links. Two nodes connected by a link represent a triplet. Depending on the community structure of the network, a node belongs to a single community C_i while the link may be of different types. If it is inside the community C_i, then it is of the type IC_i. However, if it connects two nodes belonging to two different communities, then it is of type BC.

We use only one type of link representing the link between two communities (BC) in order to minimize the possible hypotheses, since the more the number of communities increases, the more the types of links connecting two communities increase too.

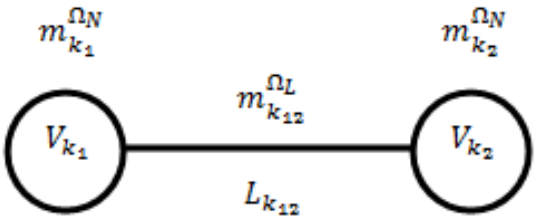

Figure 3.1: Triplet k.

In what follows, we present the general idea of the proposed method.

3.2.2 Formalization

In this work, a coherent triplet is considered, according to (Ben Dhaou et al., 2014), as a triplet $(V_{k_1}, L_{k_{12}}, V_{k_2})$ that satisfies one of the following possibilities:

- $V_{k_1} \in C_i$, $V_{k_2} \in C_i$, $L_{k_{12}} \in IC_i$ with $i = 1, \ldots N$
- $V_{k_1} \in C_i$, $V_{k_2} \in C_j$, $L_{k_{12}} \in BC$ with $(i \neq j)$, and $i, j = 1, \ldots N$

Figure 3.1 shows the notations for a given triplet k. It consists of two nodes (starting node, arrival node and link that connects them) having each one a mass function which shows the belonging possibilities of a node to a community C_i. Nodes are connected through a link, that also has a mass function which indicates the possibilities of its label (A link can be of the type IC_i if it is inside the community or BC if it connects two nodes belonging to two different communities).

Thus, the triplet is defined as follow:

- V_{k_1} modelised with a mass function $m^{\Omega_N}_{k_1}$
- V_{k_2} modelised with a mass function $m^{\Omega_N}_{k_2}$
- $L_{k_{12}}$ modelised with a mass function $m^{\Omega_L}_{k_{12}}$

We remind that a categorical mass function is a mass function with an unique focal element such that $m^{\Omega}(A) = 1$. The representatives below represent the community centres. The distances between the mass functions of the nodes and links

and categorical mass functions of the representatives are calculated in order to be able to place these elements in a group.

- For the nodes: the categorical mass functions are defined by $m_{\omega}^{\Omega_N}(\omega) = 1$ with $\omega \in \Omega_N$, *i.e.* $m_{C_i}^{\Omega_N}(C_i) = 1$, with $i = 1, \ldots, N$.
- For the links: the categorical mass functions are defined by $m_{\omega}^{\Omega_L}(\omega) = 1$ with $\omega \in \Omega_L$, *i.e.* $m_{BC}^{\Omega_L}(BC) = 1$ or $m_{IC_i}^{\Omega_L}(IC_i) = 1$, with $i = 1, \ldots, N$.

The aim of the proposed approach is to correct the noise added to a network by considering each triplet independently of the others. To do this, our algorithm proceeds by calculating the distances between the mass functions of each element of the triplet and the mass functions of the representatives of the communities. Then, it calculates the average distances of the 3 elements of the triplet and compares them with the average distances of the coherent triplets defined initially. The algorithm then keeps the minimum average distance which gives us an idea about the type of the triplet.

The value of this minimum average distance is considered as a mass function from the current information of the network and is combined thereafter with the initial mass functions. Subsequently, for each node with several links, we combine with the mean rule all the mass functions that are related to it. Finally, we use the pignistic probability to make a decision about the membership of a node to a community and a link to a given type.

The main steps of the proposed approach are detailed in what follows.

3.2.3 Main Steps of the Algorithm

The proposed approach is applied in 4 steps detailed below. We present in the following the equations used in one iteration t of the algorithm.

Step 1:
For each element of a triplet k, the distances between the latter and the corresponding categorical mass functions are calculated.

In the theory of belief functions, a distance can be used to describe the difference between two distinct sources of information. The distance of Jousselme

which takes into account the quantification of the similarity between the focal elements using Jaccard similarity coefficients is used.

By calculating the distance between the mass function of a node or a link and the corresponding categorical mass functions that are "ideals", we have an idea about its belonging to a community or a kind of link. In fact, we keep the minimum distance and the decision corresponds to the categorical mass functions having the lowest distance with the mass function of the nodes or of the links. Hence, for each triplet $(V_{k_1}, L_{k_{12}}, V_{k_2})$, with $k = 1, \ldots, M$, M the number of triplets (or links) we calculate at iteration t:

$$C_{k_1} = \arg\min_{\omega \in \Omega_N} d_J(m_{k_1}^{\Omega_N}, m_{\omega}^{\Omega_N}) \tag{3.1}$$

$$C_{k_2} = \arg\min_{\omega \in \Omega_N} d_J(m_{k_2}^{\Omega_N}, m_{\omega}^{\Omega_N}) \tag{3.2}$$

L_{k12} is determined according to the coherent triplets by:

$$L_{k_{12}} = \begin{cases} IC_{k_1} \text{ if } C_{k_1} = C_{k_2} \\ BC \text{ if } C_{k_1} \neq C_{k_2} \end{cases} \tag{3.3}$$

Table 3.1 shows the coherent values of a triplet for the case of a network containing 3 communities. This process of decision is given by (Essaid et al., 2014).

Step 2:

For each triplet k, at the iteration t we calculate the average distance d_k obtained from each possible combination presented previously.

Hence, d_k represents a minimal distance between the triplet k and the most possible categorical triplet. This average distance makes it possible to calculate the dissimilarity between any triplet and another coherent one defined initially. It is defined by:

$$d_k = \frac{d_J(m_{k_1}^{\Omega_N}, m_{C_{k_1}}^{\Omega_N}) + d_J(m_{k_{12}}^{\Omega_L}, m_{L_{k_{12}}}^{\Omega_L}) + d_J(m_{k_2}^{\Omega_N}, m_{C_{k_2}}^{\Omega_N})}{3} \tag{3.4}$$

Step 3: Knowledge Review

In this step, we use the obtained value of the average distance d_k to define a mass

function, that will be combined with the initial mass functions of the nodes and links composing each triplet. Therefore, the average distance d_k value is assigned to the focal elements that represent the types of the two nodes and the link composing the triplet k and the rest is assigned to the ignorance. Hence, we have:

$$\begin{cases} m^{\Omega_N}_{k_{1d}}(C_{k_1}) = 1 - d_k \\ \quad m^{\Omega_N}_{k_{1d}}(\Omega_N) = d_k \end{cases} \tag{3.5}$$

$$\begin{cases} m^{\Omega_L}_{k_{12d}}(L_{k_{12}}) = 1 - d_k \\ \quad m^{\Omega_L}_{k_{12d}}(\Omega_L) = d_k \end{cases} \tag{3.6}$$

$$\begin{cases} m^{\Omega_N}_{k_{2d}}(C_{k_2}) = 1 - d_k \\ \quad m^{\Omega_N}_{k_{2d}}(\Omega_N) = d_k \end{cases} \tag{3.7}$$

Once the minimum average distance has been found, we know to which coherent triplet initially defined, the current triplet k is the closest. Therefore, the nature of each of its elements is known. Hence, we know if the link which connects the two nodes is of type IC_i or BC.

The minimum average distance d_k is an information provided by a network whose initial mass functions can be noisy. Therefore, this should be taken into account when reviewing knowledge.

Calculation of final Mass Functions

In this step, we update at the iteration $t + 1$ the mass functions obtained from the previous step with the initial mass functions given at the iteration t by the following equations:

$$m^{t+1,\Omega_N}_{k_1} = m^{t,\Omega_N}_{k_1} \oplus m^{t,\Omega_N}_{k_{1d}} \tag{3.8}$$

$$m^{t+1,\Omega_L}_{k_{12}} = m^{t,\Omega_L}_{k12} \oplus m^{t,\Omega_L}_{k_{12d}} \tag{3.9}$$

$$m^{t+1,\Omega_N}_{k_2} = m^{t,\Omega_N}_{k_2} \oplus m^{t,\Omega_N}_{k_{2d}} \tag{3.10}$$

$m^{t,\Omega_N}_{k_{1d}}$, $m^{t,\Omega_L}_{k_{12d}}$, $m^{t,\Omega_N}_{k_{2d}}$ are given respectively by equations (3.5), (3.6) and (3.7).

The combination of the mass functions derived from the minimal average distance calculation and the initial generation by the Dempster rule provides a final idea of nodes and links belonging to their clusters. The Dempster rule affects the generated conflict to the focal elements and therefore there is no mass associated with the empty set.

Step 4:

As each triplet is treated independently of the others, it is possible to have cases where several links start from the same node and thus the same node can have several mass functions. In order to determine an unique mass function for each node (*e.g.* V_{k_1}), we combine by the mean rule (given by equation (1.32)), all the mass functions obtained for the given node V_{k_1} in step 3 (equation (3.9)). The choice of the mean is due to the fact that mass functions are dependent. Hence, for a given node V_{k_1}, with M_{k_1} links, we modify the mass functions by:

$$m_{k_1}^{\Omega_N} = \frac{1}{|T|} \sum_{\{k:V_{k_1} \in T\}} m_k^{\Omega_N} \tag{3.11}$$

where $T = \{(V_{k_1'}, L_{k_{12}}, V_{k_2})\}$ represents the triplets that contain the node V_{k_1} and $m_k^{\Omega_N}$ is given by the equation (3.8).

Finally, the *BetP* given by equation (1.22) is used to make decision about the belonging of the triplet $(V_{k_1}, L_{k_{12}}, V_{k_2})$. We have at the iteration $t+1$, in the order of the triplet:

$$C_{k_1} = \arg\max_{X \in \Omega_N} \sum_{Y \in 2^{\Omega_N}, Y \neq \emptyset} \frac{|\, X \cap Y \,|}{|\, Y \,|} m_{k_1}^{\Omega_N}(Y) \tag{3.12}$$

$$L_{k_{12}} = \arg\max_{X \in \Omega_L} \sum_{Y \in 2^{\Omega_L}, Y \neq \emptyset} \frac{|\, X \cap Y \,|}{|\, Y \,|} m_{k_{12}}^{\Omega_L}(Y) \tag{3.13}$$

$$C_{k_2} = \arg\max_{X \in \Omega_N} \sum_{Y \in 2^{\Omega_N}, Y \neq \emptyset} \frac{|\, X \cap Y \,|}{|\, Y \,|} m_{k_2}^{\Omega_N}(Y) \tag{3.14}$$

Algorithm 3.1 shows the outline of the process followed for correcting noise in social network using evidential attributes.

The use of the Dempster combination rule makes it possible to reinforce from one iteration to another the mass values of the elements on which the sources agree. Indeed, if we have a mass coming from each source on the same focal element, the combination rule of Dempster allows to increase the belief on the latter. From the fact that the Dempster combination rule has property of reinforcing the belief on the focal elements with which most of the sources agree, there is no change in the decision. Hence, it can be confirmed that the proposed method is still converging to a single element by the decision process given by equations (3.12), (3.13) and (3.14).

Algorithm 3.1 An Evidential Approach for Correcting Noise

Require: Graph $G(V, E)$, The set of labelled nodes, the set of labelled links
Ensure: The corrected graph
$t = 0$
repeat
1. for each element of a triplet k, compute the distance of Jousselme between the mass function of the element and the corresponding categorical mass functions using Eqs (3.1), (3.2), (3.3)
2. for each triplet k, compute the minimum average distance d_k by using Eq (3.4)
3. Define mass functions from the computed d_k using the Eqs (3.5), (3.6), (3.7)
4. Update the mass functions using the Eqs (3.8),(3.9), (3.10),
5. Combine the mass functions for the same node in order to have a unique mass function by using the Eq (3.11)
6. Make decision about the belonging of each element of the triplet k using Eqs (3.12), (3.13), (3.14)
7. $t = t + 1$
until The results of Eqs (3.12), (3.13) and (3.14) are stable.

3.3 Experiments

3.3.1 Process of Experiments

Experiments start with the generation of mass functions on the nodes and links according to the structure of the network. Indeed, for each node belonging to C_i, two focal elements are generated: one on C_i and the second one on Ω_N and the highest generated value is assigned to C_i. The same process is applied for the links: depending on the type of the link, two focal elements are generated.

In a second step, the network is noised according to three scenarios:

- **Noisy Nodes Only:** In this case, a certain number of nodes of the initial network are selected randomly and their mass functions are modified by randomly generating two focal elements (ignorance and another element except the empty set).
- **Noisy Links Only:** In this case, a certain number of links of the initial network are selected randomly and their mass functions are modified by randomly generating two focal elements (ignorance and another element except the empty set).
- **Noisy Nodes and Noisy Links:** In the latter case, some nodes and links of the networks are selected randomly. Then, their mass functions are modified.

After that, for each triplet, the distances between the attributes of the link as well as the two nodes and the attributes of the representatives are calculated. As different networks with N communities are considered, the coherent triplets are defined on the basis of the community structure of the networks. That is to say, a node can belong to only one community C_i. From this hypothesis, the links that we can have are of type IC_i if they are inside the community C_i, if not the links are of type BC (if the nodes belong to two different communities).

Then, we calculated the average of the distances of the elements composing the triplet based on the possibilities defined initially. Table 3.1 presents the possible triplets for the case of a network of 3 communities.

Thereafter, the kept minimum average distance is combined with the initial mass functions by the Dempster rule. Here, the initial mass functions represent

V_{k1}	V_{k2}	L_{k12}
C_1	C_1	IC_1
C_1	C_2	BC
C_1	C_3	BC
C_2	C_2	IC_2
C_2	C_1	BC
C_2	C_3	BC
C_3	C_3	IC_3
C_3	C_1	BC
C_3	C_2	BC

Table 3.1: Coherent Triplets For 3 Communities.

the mass functions before the calculation of our model is applied. For each node V_{ki} belonging to several triplets, all the mass functions obtained at the end of the calculation of the Dempster combination are combined by the mean rule.

The proposed algorithm is iterative since, for several cases of noisy nodes and/or noisy links, the corrections are made only after a certain number of iterations.

The mass functions obtained at the end of each iteration represent the input of the next iteration. For each iteration, we calculated the confusion matrix. The confusion matrix is a technique for summarizing the performance of a classification algorithm.

In order to know the accuracy value at each iteration for each case to be tested, we compared the result of the pignistic probability applied at the end of each iteration with the initial information of the network before introducing the noise. The accuracy represents the ratio of correct predictions to total predictions made.

In order to show the efficiency of our method, we compare the obtained results with those of the baseline. All experiments were repeated 10 times for cross validation. All figures represent the average of the accuracy calculated for 10 runs. In addition, the evidential approach and the probabilistic one are tested under the same conditions: The same elements randomly selected and noisy in the evidential case are noisy during the probabilistic approach test.

In the tables presented in the following, we present the accuracy averages as

well as the confidence intervals obtained from the evidential approach and the baseline for each type of experiment.

3.3.2 Possible Corrections

In the presence of noise, the algorithm corrects the information of the network as a function of the noisy elements and the coherent triplets initially defined. In this section, we present the possible corrections for the case of a network containing 3 communities:

One noisy node and the link and the other node are corrects Initially the triplet: $V_{k1} \in C_1$, $L_{k12} \in IC_1$, $V_{k2} \in C_1$ is considered. Suppose that one of the nodes is modified and belongs now to C_2 or C_3. The algorithm will detect that according to the information given by the link and the other node, the modified one should be corrected. Therefore, the noisy node will be affected to C_1. It is the same if we have a triplet $V_{k1} \in C_2$, $V_{k2} \in C_2$, $L_{k12} \in IC_2$ or a triplet $V_{k1} \in C_3$, $V_{k2} \in C_3$, $L_{k12} \in IC_3$. The noisy node will be reassigned to its initial community.

Two noisy nodes and the link is correct In that case, the algorithm will change the nature of the link to obtain a coherent triplet. If the modified nodes belongs to the same community, the algorithm will change the link in such a way that it will be internal to the same community. If the modified nodes belongs to different communities, the algorithm will change the nature of the link to "Between Clusters" (BC).

One noisy node, one noisy link and one correct node Suppose that initially we had, $V_{k1} \in C_1$, $L_{k12} \in IC_1$ and $V_{k2} \in C_1$. V_{k1} was modified to belong to C_2 or C_3, $L_{k12} \in BC$ and $V_{k2} \in C_1$. In that case, the algorithm will not change the information of the triplet because it's coherent. However, if we have for example $V_{k1} \in C_2$ or C_3, $L_{k12} \in IC_2$ *or* IC_3 and $V_{k2} \in C_1$, the algorithm will change the link to BC and if one of the nodes (or both) are connected to other nodes, so the algorithm will have another information and can change one of the node based on that.

Two noisy nodes and noisy link In that case, the algorithm will compute the minimal distance between the current triplet and the coherent ones defined initially and then modify the information of the current triplet.

3.3.3 Convergence

The previous presented algorithm is iterative which allows to obtain better results of the accuracy from one iteration to another. The stop criterion used is the stabilization of the value of the accuracy.

In these experiments the algorithm is performed for only 5 iterations since beyond this number, the variation of the accuracy becomes negligible.

In order to show the convergence of our evidential approach, an LFR network composed of 99 nodes, 191 links and 3 communities is considered. The noise is added to 30 nodes and 50 links and the behaviour of the proposed algorithm is evaluated.

Figure 3.2 shows the evolution of the accuracy from an iteration to another. The case of 30 noisy nodes and 50 noisy links is tested (Evidential Attributes). It can be noticed that from an iteration to another, the accuracy value increases which means that the algorithm succeeds in correcting the noise.

3.3.4 Baseline

In order to show the efficiency of the proposed method, we have performed an algorithm that uses the same principle in probabilistic version. A method of literature wasn't used since, to our knowledge, there is no work that has considered the resolution of the same problem. Figure 3.3 presents a probabilistic triplet. For each node and the link connecting them a vector of probabilities is associated.

Step 1: Generation of Probabilities

In this step, N values in [0, 1] for each node and $N+1$ probabilities for each link are generated then we normalize. $N+1$ probabilities are generated as we have IC_i links within communities and BC links that connect communities to each other. Then, the maximum generated probability is associated with the class to which the node/link belongs. The vector of probabilities is defined as follow:

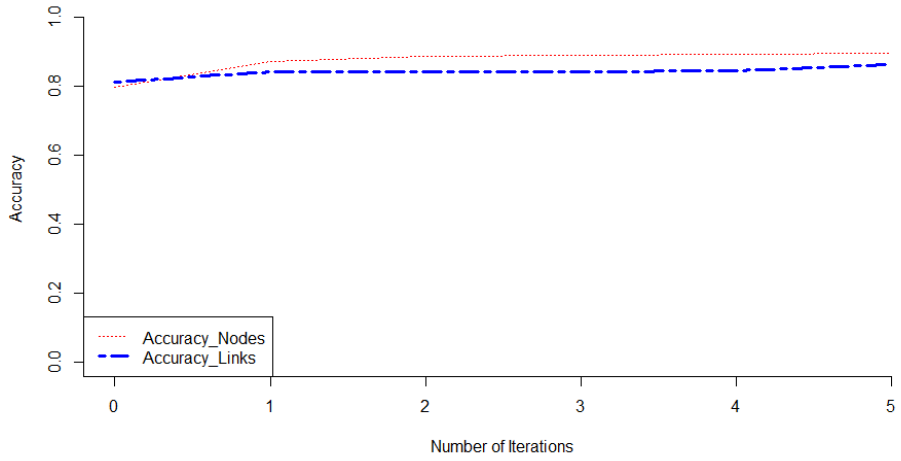

Figure 3.2: LFR: corrected nodes and links: case of 30 noisy nodes and 50 noisy links.

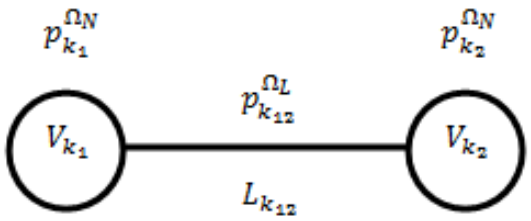

Figure 3.3: Probabilistic Triplet.

- $(p(C_1), p(C_2), ..., p(C_N))$ for each node.
- $(p(IC_1), p(IC_2), p(IC_3), ..., p(IC_N), p(BC))$ for each link.

Step 2: Calculation of Distances

In this step, the Euclidean distances between the attributes of each node/link composing a triplet with those of the representatives of each group are calculated:

- For the nodes: The probabilities on certain events are defined by $p^{\Omega_N}_{\omega}(\omega) = 1$ with $\omega \in \Omega_N$ *i.e.* $p^{\Omega_N}_{C_i}(C_i) = 1$, with $i = \{1, ..., N\}$.
- For the links: The probabilities on certain events are defined by $p^{\Omega_L}_{\omega}(\omega) = 1$ with $\omega \in \Omega_L$ *i.e.* $p^{\Omega_L}_{BC}(BC) = 1$ or $p^{\Omega_L}_{IC_i}(IC_i) = 1$, with $i = \{1, ..., N\}$.

Depending of the number of communities composing the network, every representative has 1 on the attribute of its class and 0 on the others. For example, if we consider a representative of C_1 and we have 3 communities in the network, its probabilities vector is $R_1 = (1, 0, 0)$.

Hence, we have:

$$C_{k_1} = \arg\min_{\omega \in \Omega_N} d_E(p_{k_1}^{\Omega_N}, p_{\omega}^{\Omega_N}) \tag{3.15}$$

$$C_{k_2} = \arg\min_{\omega \in \Omega_N} d_E(p_{k_2}^{\Omega_N}, p_{\omega}^{\Omega_N}) \tag{3.16}$$

$L_{k_{12}}$ is determined according to the coherent triplets by:

$$L_{k_{12}} = \begin{cases} IC_{k_1} \text{ if } C_{k_1} = C_{k_2} \\ BC \text{ if } C_{k_1} \neq C_{k_2} \end{cases} \tag{3.17}$$

Step 3: Calculation of Average Distances

In this step, the minimal average distance of each triplet k is calculated as follow:

$$d_k = \frac{d_E(p_{k1}^{\Omega_N}, p_{C_{k_1}}^{\Omega_N}) + d_E(p_{k_{12}}^{\Omega_L}, p_{L_{k_{12}}}^{\Omega_L}) + d_E(p_{k_2}^{\Omega_N}, p_{C_{k_2}}^{\Omega_N})}{3} \tag{3.18}$$

Step 4: Assignment of probabilities from distances

In this step, the probabilities resulting from the computation of the distances between triplets are assigned. The values of the minimal average distance d_k are used.

Hence, we have:

$$\begin{cases} p_{k_{1d}}^{\Omega_N}(C_{k_1}) = 1 - d_k \\ p_{k_{1d}}^{\Omega_N}(\overline{C_{k_1}}) = d_k \end{cases} \tag{3.19}$$

$$\begin{cases} p_{k_{12d}}^{\Omega_L}(L_{k_{12}}) = 1 - d_k \\ p_{k_{12d}}^{\Omega_L}(\overline{L_{k_{12}}}) = d_k \end{cases} \tag{3.20}$$

$$\begin{cases} p_{k_{2d}}^{\Omega_N}(C_{k_2}) = 1 - d_k \\ p_{k_{2d}}^{\Omega_N}(\overline{C_{k_2}^t}) = d_k \end{cases} \tag{3.21}$$

where $\overline{C_{k_1}}, \overline{L_{k_{12}}}, \overline{C_{k_2}}$ represent respectively the elements contrary to $C_{k_1}, L_{k_{12}}, C_{k_2}$.

Step 5: Calculation of the average between the new probabilities and the initial ones

In order to have a single probability distribution for each node/link, the average between the probabilities generated in the first instance and those resulting from the calculation of the distances is calculated.

$$p_{k_1}^{t+1,\Omega_N} = \frac{p_{k_1}^{t,\Omega_N} + p_{k1d}^{t,\Omega_N}}{2} \tag{3.22}$$

$$p_{k_{12}}^{t+1,\Omega_L} = \frac{p_{k_{12}}^{t,\Omega_L} + p_{k12d}^{t,\Omega_L}}{2} \tag{3.23}$$

$$p_{k_2}^{t+1,\Omega_N} = \frac{p_{k_2}^{t,\Omega_N} + p_{k2d}^{t,\Omega_N}}{2} \tag{3.24}$$

where $p_{k_{1d}}^{t,\Omega_N}$, $p_{k_{12d}}^{t,\Omega_L}$, $p_{k_{2d}}^{t,\Omega_N}$ are given respectively by equations (3.19), (3.20) and (3.21).

In order to determine a unique probabilities vector for each node (*e.g.* V_{k_1}), all the probabilities obtained for the given node V_{k_1} are combined by the mean rule (given by equation (1.32)). Hence, we have:

$$p_{k1}^{\Omega_N} = \frac{1}{|T|} \sum_{\{k:V_{k_1} \in T\}} p_k^{\Omega_N} \tag{3.25}$$

where $T = \{(V_{k'_1}, L_{k_{12}}, V_{k_2})\}$ and $p_k^{\Omega_N}$ is given by the equation (3.22).

Step 6: Making Decision

In this step, the membership of each node/link is decided. To do this, we decide the singleton having the maximum of probability.

Algorithm 3.2 shows the outline of the process followed for correcting noise in social network using probabilistic attributes.

In order to test the effectiveness of the baseline, the noise is added as it was done with the evidential approach. To do this, the noise is added to the same nodes and links selected randomly when the evidential approach is tested.

Algorithm 3.2 A Probabilistic Approach for Correcting Noise

Require: Graph $G(V, E)$, The set of labelled nodes, the set of labelled links
Ensure: The corrected graph.
$t = 0$
repeat
1. for each element of a triplet k, compute the Euclidean distance between the element and the corresponding categorical representative using Eqs (3.15), (3.16), (3.17)
2. for each triplet k, compute the minimum average distance d_k by using Eq (3.18)
3. Define probabilities from the computed d_k using the Eqs (3.19), (3.20), (3.21)
4. Update the probabilities using the Eqs (3.22),(3.23), (3.24),
5. Combine the probabilities for the same node in order to have a unique vector of probabilities by using the Eq (3.25)
6. Make decision about the belonging of each element of the triplet k
7. $t = t + 1$
until Number of iterations equal to 5.

3.3.5 Improvement Rate

Tables 3.2, 3.3, 3.4, 3.5 show the rate of improvement of the evidential approach compared to the baseline at the fifth iteration. The variation of noise in the LFR network composed of 99 nodes, 191 links and 3 communities is considered.

The rate of improvement is calculated by making the difference between the average values of the accuracy obtained with the evidential approach at the fifth iteration with that given by the baseline.

Noise	Rate of improvement
30 Nodes	60%
60 Nodes	53%
90 Nodes	42%
99 Nodes	38%

Table 3.2: Improvement Rate: Case of Noisy Nodes Only.

Noise	Rate of improvement
50 Links	41%
100 Links	36.7%
191 Links	36%

Table 3.3: Improvement Rate: Case of Noisy Links Only.

Noise	Rate of improvement
30 Nodes + 50 Links	45%
60 Nodes + 100 Links	32%
90 Nodes + 191 Links	11%
99 Nodes + 191 Links	7%

Table 3.4: Improvement Rate for Nodes: Case of Noisy Nodes and Noisy Links.

Noise	Rate of improvement
30 Nodes + 50 Links	50%
60 Nodes + 100 Links	27%
$90Nodes$ + 191 Links	6%
99 Nodes + 191 Links	4%

Table 3.5: Improvement Rate for Links: Case of Noisy Nodes and Noisy Links.

Noise	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
10 Nodes	0.9265	[0.911, 0.941]	0.51471	[0.443, 0.585]
20 Nodes	0.86469	[0.807, 0.922]	0.50589	[0.422, 0.588]
30 Nodes	0.7647	[0.683, 0.845]	0.45882	[0.328, 0.589]
34 Nodes	0.7558	[0.634, 0.876]	0.4076	[0.313, 0.565]

Table 3.6: Accuracy Average and Interval of Confidence: Case of Noisy Nodes Only in the Karate Club.

3.3.6 Experiments on Real Data: Karate Club

As the karate club network has 2 communities, the frames of discernment of the nodes and links are defined by:

- $\Omega_N = \{C_1, C_2\}$
- $\Omega_L = \{IC_1, IC_2, BC\}$

In this part, the results obtained in the case of noisy nodes only, noisy links only and noisy nodes and links at the same time are shown.

Noisy Nodes Only

In figure 3.4 we present the accuracy average values at the fifth iteration when we vary the number of noisy nodes.

It is noticed that the more the number of noisy nodes increases, the more the accuracy average value decreases for both evidential and probabilistic methods.

However, it is remarked that we obtain a better accuracy average results with the theory of belief functions comparing to the probability theory. This can be explained by the fact that the theory of belief functions manages ignorance as well as conflict.

Table 3.6 presents the accuracy averages and the confidence intervals obtained from the evidential approach and the baseline for each level of noise added to the nodes only in the case of the Karate Club.

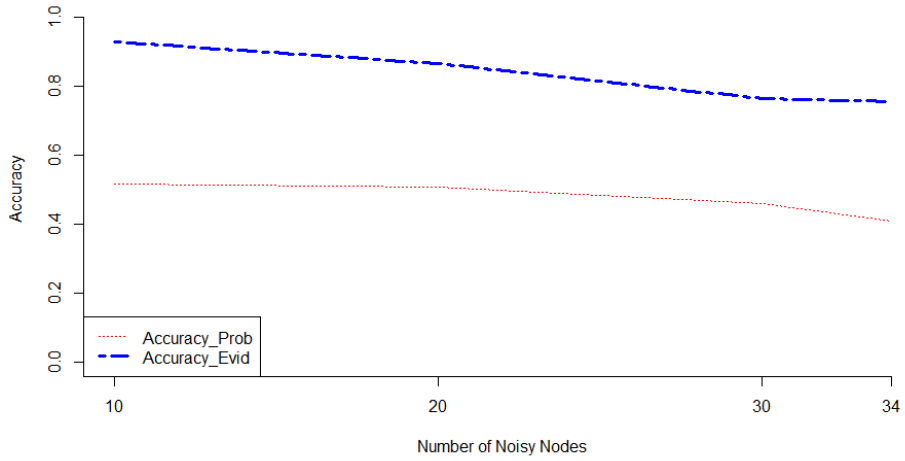

Figure 3.4: Karate Club: comparison of probabilistic and evidential accuracy: case of noisy nodes.

Noise	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
20 Links	0.94225	[0.923, 0.960]	0.66665	[0.629, 0.703]
40 Links	0.88975	[0.854, 0.924]	0.63333	[0.569, 0.696]
60 Links	0.80771	[0.762, 0.852]	0.60128	[0.564, 0.637]
78 Links	0.76538	[0.704, 0.826]	0.56922	[0.529, 0.608]

Table 3.7: Accuracy Average and Interval of Confidence: Case of Noisy Links Only in the Karate Club.

Noisy Links Only

Figure 3.5 shows the accuracy average results at the fifth iteration after noising 20, 40, 60 and 78 links of the network.

According to the curve, the average accuracy value given by the evidential approach is better than that given by the baseline in each level of noise.

Table 3.7 presents the obtained accuracy averages and the confidence intervals given by the evidential method and the probabilistic approach when the number of noisy links only is varied in the case of the Karate Club.

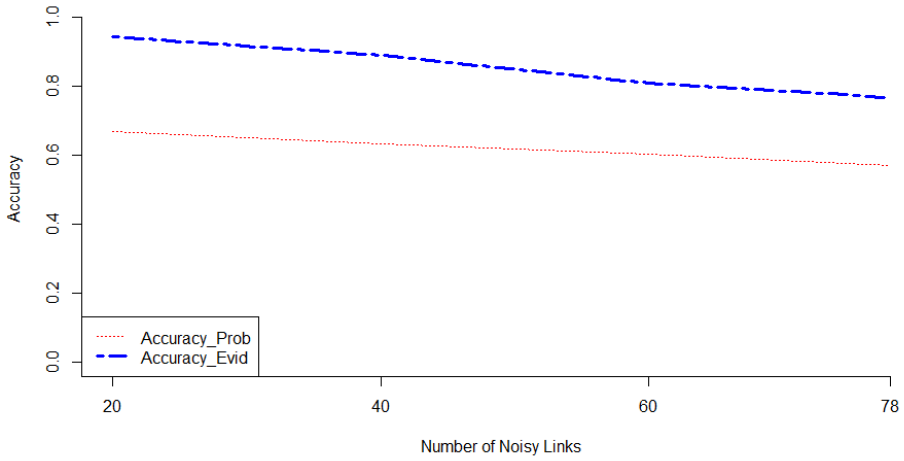

Figure 3.5: Karate Club: comparison of probabilistic and evidential accuracy: case of noisy links.

Noisy Nodes and Noisy Links

In this third case, we proceed by noising the nodes and the links at the same time. Figure 3.6 shows the obtained results of accuracy average after noising the attributes at the fifth iteration. The abscissa represents respectively the level of noise 10 nodes and 20 links, 20 nodes and 40 links, 30 nodes and 60 links and finally, 34 nodes and 78 links.

It is noticed that the accuracy average values decreases as the noise level increases for both evidential and probabilistic approaches. However, the proposed method gives better results than the baseline.

Table 3.8 shows the obtained accuracy averages and the confidence intervals given by the evidential method and the probabilistic approach in the case of noisy nodes and noisy links in the case of the Karate Club.

3.3.7 Experiments on LFR

In the second part of the experiments, different networks generated with LFR benchmark are used. The parameters used to generate our networks are presented in Appendix A.

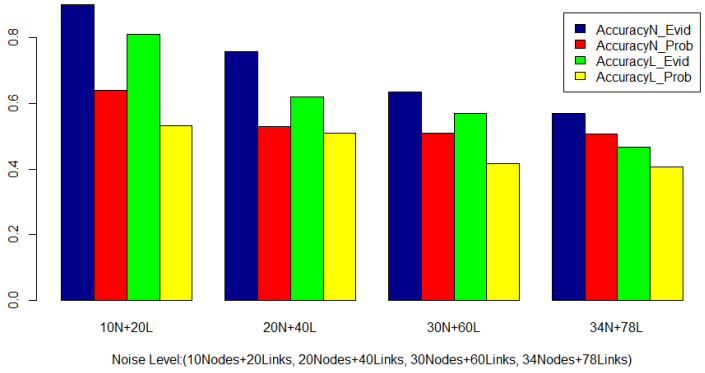

Figure 3.6: Karate Club: comparison of probabilistic and evidential accuracy: case of noisy nodes and links.

Several performed experimentations are repeated 10 times and the obtained averages of the accuracy are presented. All the figures present the results given by the evidential approach and the baseline.

First, the noise is added to the nodes, links and both of them in the case of the LFR network composed of 99 nodes, 191 links and 3 communities.

For the rest of the experiments, each time one of the parameters of the LFR network is varied such as the number of communities, the size of the network as well as the mixing parameter μ and their impact on the noise correction rate is observed. For each of these experiments we noise 60% of the nodes and 50% of the links.

The first set of experiments consists of varying the noise in an LFR network composed of 99 nodes, 191 links and 3 communities. We proceed by noising the nodes at first, then the links and finally we simultaneously noise both.

The frames of discernment of the nodes and links for this network are defined as follows:

- $\Omega_N = \{C_1, C_2, C_3\}$
- $\Omega_L = \{IC_1, IC_2, IC_3, BC\}$ with IC_i represents the links inside the community C_i and BC represents the links between 3 communities.

Case of Nodes				
Noise	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
10 Nodes+ 20 Links	0.90004	[0.871, 0.928]	0.63972	[0.581, 0.697]
20 Nodes+ 40 Links	0.758228	[0.689, 0.827]	0.52949	[0.467, 0.591]
30 Nodes+ 60 Links	0.6353	[0.559, 0.711]	0.50833	[0.439, 0.578]
34 Nodes+ 78 Links	0.56882	[0.449, 0.667]	0.50589	[0.395, 0.616]
Case of Links				
Noise	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
10 Nodes+ 20 Links	0.81026	[0.738, 0.882]	0.53234	[0.394, 0.669]
20 Nodes+ 40 Links	0.61922	[0.534, 0.703]	0.50883	[0.445, 0.598]
30 Nodes+ 60 Links	0.56882	[0.483, 0.638]	0.41538	[0.329, 0.5011]
34 Nodes+ 78 Links	0.465614	[0.383, 0.528]	0.40641	[0.359, 0.453]

Table 3.8: Accuracy Average and Interval of Confidence: Case of Noisy Nodes and Links in the Karate Club.

Noisy Nodes Only

In this first case of experiments, the noise is added to a number of nodes randomly selected of the network. The noise consists on modifying the mass functions of the selected nodes by randomly generating two focal elements (ignorance and another element except the empty set). Then, the obtained results are compared with those given by the baseline. Figure 3.7 shows the obtained results of the accuracy for every variation of the noise. The number of noisy nodes is varied from 30 to 99.

It is noticed that the more the number of noisy nodes increases the more the accuracy average decreases. The evidential model gives better results than the baseline. This is because the theory of belief functions offers a very effective way to handle ignorance and conflict.

Table 3.9 shows the obtained accuracy averages and the confidence intervals given by the evidential method and the probabilistic approach in the case of noisy nodes only in the case of LFR network.

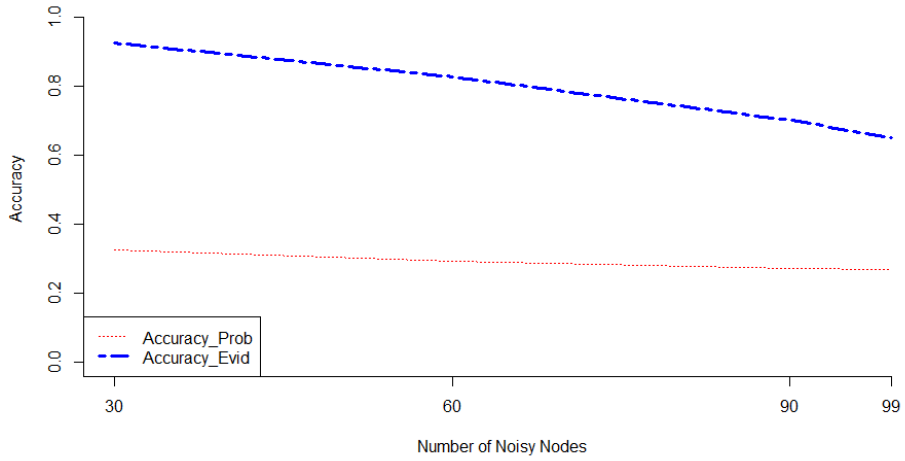

Figure 3.7: LFR: comparison of probabilistic and evidential accuracy: case of noisy nodes.

Noise	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
30 Nodes	0.92526	[0.894, 0.955]	0.32522	[0.267, 0.383]
60 Nodes	0.82729	[0.781, 0.873]	0.29391	[0.266, 0.321]
90 Nodes	0.70205	[0.622, 0.781]	0.2727	[0.258, 0.298]
99 Nodes	0.65054	[0.610, 0.690]	0.26866	[0.244, 0.292]

Table 3.9: Accuracy Average and Interval of Confidence: Case of Noisy Nodes Only in LFR.

Noisy Links Only

The second part of the experiments consists in keeping the initial generation of the mass functions of the nodes and adding noise only to the mass functions of the links.

Figure 3.8 shows the obtained results of the accuracy average due to the variation in the number of noisy links. In this figure, we compute the accuracy average for 50, 100 and 191 noisy links. The proposed approach gives better results than the probabilistic one. These results can be explained by the fact that the evidential approach better manages ignorance than the probabilistic approach.

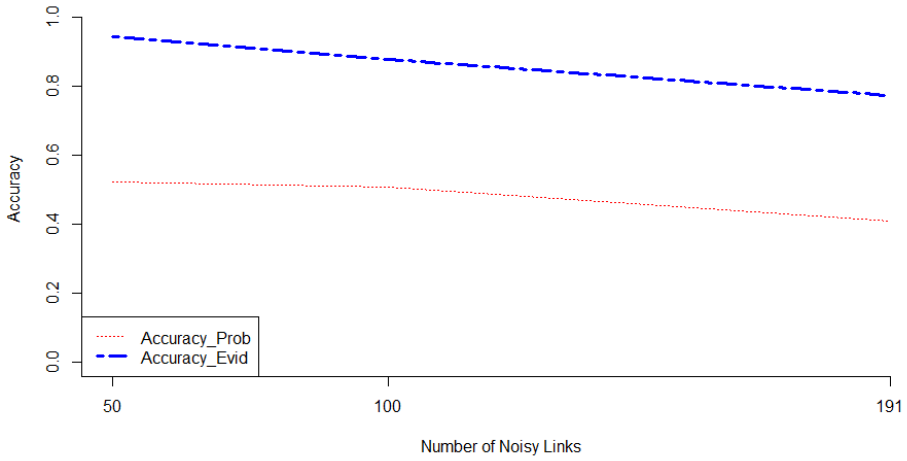

Figure 3.8: LFR: comparison of probabilistic and evidential accuracy: case of noisy links.

Noise	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
50 Links	0.94239	[0.930, 0.953]	0.52252	[0.474, 0.570]
100 Links	0.87539	[0.862, 0.887]	0.50786	[0.458, 0.557]
191 Links	0.77119	[0.739, 0.803]	0.40988	[0.352, 0.467]

Table 3.10: Accuracy Average and Interval of Confidence: Case of Noisy Links Only in LFR.

Table 3.10 presents the accuracy averages and the confidence intervals obtained from the evidential approach and the baseline in the case of noisy links only in the case of LFR network.

Noisy Nodes and Noisy Links

In this third part of the experiments, the nodes and links are noised simultaneously.

The aim of simultaneously noising the nodes and the links is to make the network totally incoherent and to evaluate the ability of the algorithms to correct the noise and to find a network comparable to the initial one.

The number of noisy nodes is varied by 30 at each step and then all the nodes

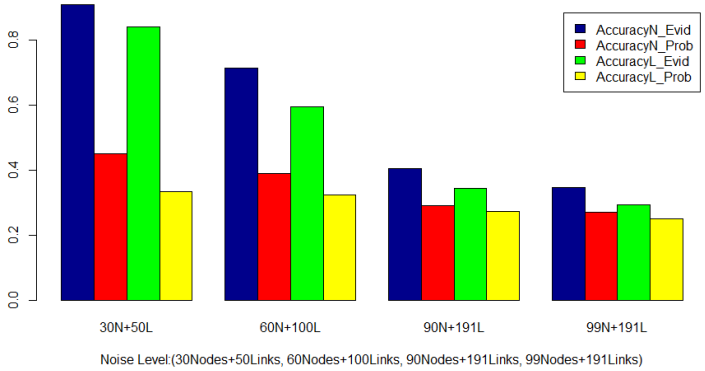

Figure 3.9: LFR: comparison of probabilistic and evidential accuracy: case of noisy nodes and links.

of the network are noised. As for the links, we vary the noisy links by 50, then the noise is added on all the links of the network.

These values are chosen in order to have a better view on the impact of the noise introduced on the network information.

The obtained results are compared with those of the baseline.

Figure 3.9 shows the results of the accuracy average for every level of noise used in these experiments. The obtained results are compared with those of the baseline after noising 30 nodes and 50 links, 60 nodes and 100 links, 90 nodes and 191 links and finally, 99 nodes and 191 links.

From this figure, it is noticed that the accuracy average results are better with the evidential attributes. It is remarked also that when it is very noisy, it becomes impossible to obtain good results.

It should be noted that in the case of adding a maximum noise, the value of the accuracy average is stable from the beginning. This is due to the fact that when we noise the data, the mass functions are generated randomly and therefore there are two possibilities:

- Either the new mass function makes sure to change the class of the node/link.
- Either the element always retains its initial membership but with a different mass function.

Case of Nodes				
Noise	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
30 Nodes+ 50 Links	0.9091	[0.882, 0.936]	0.45125	[0.390, 0.511]
60 Nodes+ 100 Links	0.71417	[0.664, 0.763]	0.3901	[0.311, 0.412]
90 Nodes+ 191 Links	0.40602	[0.367, 0.444]	0.29088	[0.245, 0.325]
99 Nodes+ 191 Links	0.34643	[0.293, 0.399]	0.27016	[0.227, 0.312]
Case of Links				
Noise	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
30 Nodes+ 50 Links	0.84188	[0.810, 0.872]	0.3333	[0.266, 0.399]
60 Nodes+ 100 Links	0.59634	[0.558, 0.633]	0.3232	[0.262, 0.383]
90 Nodes+ 191 Links	0.3434	[0.313, 0.398]	0.27436	[0.247, 0.305]
99 Nodes+ 191 Links	0.2929	[0.258, 0.312]	0.24987	[0.228, 0.275]

Table 3.11: Accuracy Average and Interval of Confidence: Case of Noisy Nodes and Noisy Links in LFR.

Hence, we always have elements that are correct even when it's the case of maximal noise. These correct attributes help in the finding of other correct triplets.

Table 3.11 presents a comparison between the accuracy averages and the confidence intervals given by the evidential approach and the baseline in the case of noisy nodes and noisy links in the case of LFR network.

In what follows, we add noise to 60% of nodes and 50% of links by varying each time a parameter of the LFR algorithm. The idea is to see the impact of each parameter on the correction rate of noisy information for the same level of noise. To do this, we first vary the N which represents the number of nodes composing the network. Then we vary the number of communities and finally, we vary the mixing parameter.

3.3.8 LFR: Variation of the Communuities Number

In this part of experiments, we vary the number of communities. We generate 4 LFR networks:

- a network with 200 nodes, 402 links and 3 communities.

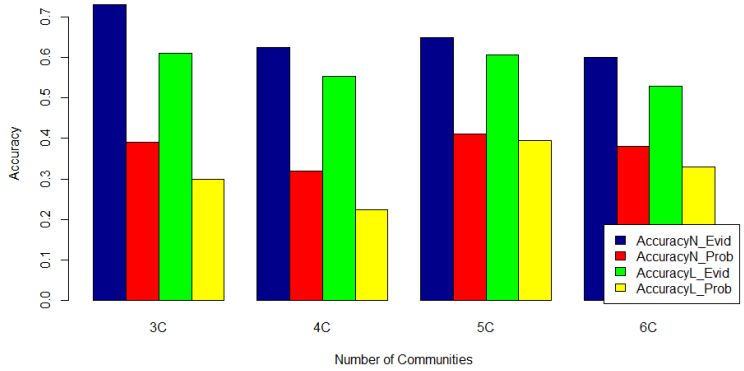

Figure 3.10: LFR: comparison of probabilistic and evidential accuracy: case of noisy nodes and links.

- a network with 200 nodes, 472 links and 4 communities.
- a network with 200 nodes, 477 links and 5 communities.
- a network with 200 nodes, 501 links and 6 communities.

In this experimentation, we modified at the same time 60% of the nodes and 50% of the links.

Figure 3.10 shows the obtained results of the accuracy average for each network. We can remark that for all the networks, the evidential model gives better results on links and nodes accuracy average than the baseline. We notice also that there is not really a big difference in the values of the accuracy average when we vary the number of communities. We can, therefore, conclude that the proposed approach is stable.

Table 3.12 presents a comparison between the accuracy averages and the confidence intervals given by the evidential approach and the probabilistic one when we vary the number of communities in the case of LFR networks.

3.3.9 LFR: Variation of the Network Size

In this section, we present the obtained results of the accuracy following the variation of the network size. We consider 5 networks whose number of nodes was varied and containing 3 communities:

Case of Nodes				
Nb-Communities	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
C3	0.73	[0.689, 0.774]	0.39	[0.321, 0.402]
C4	0.625	[0.602, 0.645]	0.32	[0.281, 0.345]
C5	0.65	[0.63, 0.679]	0.41	[0.385, 0.445]
C6	0.6	[0.598, 0.621]	0.38	[0.365, 0.4]
Case of Links				
Nb-Communities	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
C3	0.61	[0.563, 0.669]	0.30	[0.298, 0.325]
C4	0.553	[0.524, 0.573]	0.2247	[0.201, 0.251]
C5	0.6065	[0.575, 0.613]	0.3939	[0.371, 0.405]
C6	0.53	[0.508, 0.554]	0.33	[0.295, 0.353]

Table 3.12: Accuracy Average and Interval of Confidence: Case of Noisy Nodes and Noisy Links-Communities Variation.

- a network with 50 nodes and 115 links.
- a network with 99 nodes and 191 links.
- a network with 200 nodes and 402 links.
- a network with 300 nodes and 721 links.
- a network with 400 nodes and 932 links.

Figure 3.11 presents the obtained accuracy average results after adding 60% of noisy nodes and 50% of noisy links. It shows that the evidential approach was able to correct more information than the baseline whatever the network considered. Moreover, Figure 3.11 shows that the evidential method is stable since the values of the precision calculated for each network are close to each other.

Table 3.13 shows the obtained accuracy averages and the confidence intervals given by the evidential approach and the probabilistic one when we vary the size of the network in the case of LFR.

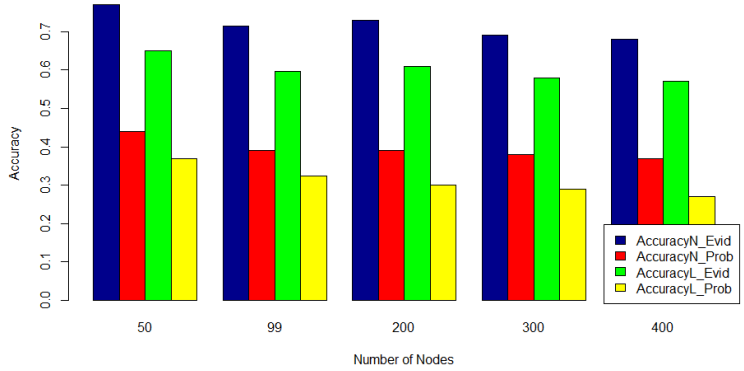

Figure 3.11: LFR: comparison of probabilistic and evidential accuracy: case of variation of the size of the network.

3.3.10 LFR: Variation of the Mixing Parameter

In this section, we present the obtained results of the accuracy average following the variation of the mixing parameter μ. We consider 5 networks whose mixing parameter was varied and containing 3 communities:

- a network with 200 nodes, 484 links and $\mu = 0.1$.
- a network with 200 nodes, and 402 links and $\mu = 0.3$.
- a network with 200 nodes, and 467 links and $\mu = 0.5$.
- a network with 200 nodes, and 488 links and $\mu = 0.7$.
- a network with 200 nodes, and 502 links and $\mu = 0.9$.

60% of the nodes and 50% of the links were noised. Figure 3.12 shows the results obtained by the evidential method and the baseline after varying the mixing parameter.

We find that the accuracy average of the nodes is greater than the accuracy average of the links when $\mu < 0.5$, while the latter becomes greater than the accuracy average of the nodes when $\mu > 0.5$. This change is explained by the fact that the more the mixing parameter approaches 1, the more we get a network with more links between clusters than within the community.

Case of Nodes				
Nb-Nodes	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
50	0.77	[0.705, 0.798]	0.44	[0.365, 0.463]
99	0.71417	[0.664, 0.763]	0.3901	[0.311, 0.412]
200	0.73	[0.698, 0.773]	0.39	[0.321, 0.402]
300	0.69	[0.602, 0.725]	0.38	[0.309, 0.395]
400	0.68	[0.598, 0.699]	0.37	[0.312, 0.385]
Case of Links				
Nb-Nodes	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
50	0.65	[0.585, 0.705]	0.37	[0.303, 0.398]
99	0.59634	[0.558, 0.633]	0.3232	[0.315, 0.3434]
200	0.61	[0.563, 0.669]	0.30	[0.298, 0.325]
300	0.58	[0.538, 0.621]	0.29	[0.205, 0.382]
400	0.57	[0.545, 0.611]	0.27	[0.203, 0.351]

Table 3.13: Accuracy Average and Interval of Confidence: Case of Noisy Nodes and Noisy Links-Network Size Variation.

Case of Nodes				
μ	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
0.1	0.732	[0.689, 0.774]	0.42346	[0.394, 0.452]
0.3	0.73	[0.687, 0.773]	0.39	[0.321, 0.402]
0.5	0.6625	[0.626, 0.698]	0.325	[0.291, 0.358]
0.7	0.645	[0.604, 0.685]	0.19939	[0.181, 0.217]
0.9	0.6315	[0.602, 0.658]	0.16455	[0.143, 0.185]
Case of Links				
μ	Evid-Accu-Av	IC-Evid	Prob-Accu-Av	IC-Prob
0.1	0.60426	[0.564, 0.644]	0.3255	[0.273, 0.377]
0.3	0.61	[0.563, 0.669]	0.30	[0.298, 0.325]
0.5	0.67687	[0.626, 0.698]	0.25868	[0.239, 0.277]
0.7	0.711	[0.690, 0.732]	0.3425	[0.320, 0.364]
0.9	0.75238	[0.741, 0.763]	0.3545	[0.3283, 0.380]

Table 3.14: Accuracy Average and Interval of Confidence: Case of Noisy Nodes and Noisy Links-Mixing Parameter Variation.

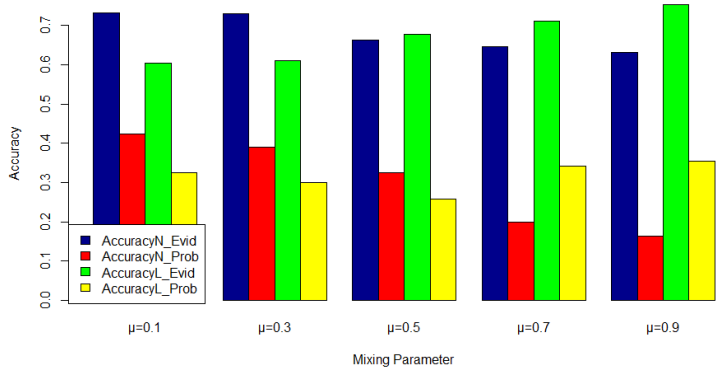

Figure 3.12: LFR: comparison of probabilistic and evidential accuracy: case of variation of the mixing parameter.

	C3	C4	C5	C6
Probabilistic Execution Time	5.45	8.1	8.95	9.45
Evidential Execution Time	119.05	652.4	3864.15	19225.4

Table 3.15: Comparison of probabilistic and evidential execution time

We present in table 3.14 the obtained accuracy averages and the confidence intervals given by the evidential approach and the baseline when we vary the mixing parameter in the case of LFR.

3.3.11 Comparison of the Execution Time

In this section, the execution time put by the model's evidential version as well as the probabilistic one are compared. The execution time at the fifth iteration is presented. The evolution of the execution time in the case of LFR networks with 6, 5, 4 and 3 communities is observed. The execution time is expressed in seconds.

Table 3.15 shows that the evidential method takes more time compared to the baseline. It is also noticed that as the number of communities increases, the execution time increases too. It is true that the evidential algorithm takes more time to give the results than the probabilistic one. However, we obtain better accuracy results with the proposed approach.

In terms of improving execution time when increasing the number of network communities, the combination rule proposed by (K. Zhou, Martin, & Pan, 2018) can be used. Indeed, it can be used to combine mass functions from a large number of sources. The Conjunctive combination Rule for a Large Number of Sources (LNS-CR) has a reasonable complexity while keeping property of reinforcing the belief on the focal elements with which most of the sources agree. Furthermore, the reliability of the sources is more relaxed, as it does not require all the sources are reliable, but only at least half of them are reliable.

3.4 Conclusion

In this chapter, we presented a method which allows to classify the nodes in their initial clusters even when there is a significant noise added to the network. In the case of a large noise, the algorithm guarantees the information coherence of any network even when it is a network whose nodes and links attributes have been strongly modified.

The proposed approach is tested on real data: the Karate Club network. Then, the noise is varied on a LFR network composed of 3 communities and the obtained results during the noising of the nodes, links and both are presented. Finally, the behaviour of the proposed method is studied according to the variation of the number of communities, the size of the network as well as the mixing parameter. All the obtained results were compared with those of the baseline.

Experiments have shown that the more noisy is the network, the more difficult it is to find the initial network. However, a coherent network is obtained. In addition, the proposed approach is stable when the number of communities and the size of the network are varied and gives better results in all studied cases than the baseline.

In the next chapter, an approach which aims to detect the spammed links based on the informations given by the nodes and links attributes in addition of the attributes associated with the messages passing through the network is introduced.

Chapter 4

A Belief Approach for Detecting Spammed Links

4.1 Introduction

Nowadays, people are interconnected whether professionally or personally using different social networks. However, we sometimes receive messages or advertisements that are not correlated to the nature of the relation established between the persons. Therefore, it became important to be able to sort out our relationships. Thus, based on the type of links that connect us, we can decide if this last is spammed and should be deleted.

In a social network, the link prediction problem aims to identify future relationships between nodes. Several link prediction techniques exist in the literature (Wang et al., 2015). They can be categorized as follows: methods that use information of nodes, methods that use topology and methods that use social theory.

The techniques using the information of the nodes are based in the idea that the more similar the pair is, the more likelihood a link between them.

Regarding the techniques using the topology, they are used when we do not have node or edge attributes. Indeed, they are based on the graph structural features.

For the case of the techniques using social theory, they use additional social

interaction information such as community, triadic closure, strong and weak ties, homophily, and structural balance.

Another interesting problem appears in social networks which is the spammers detection problem. We find several approaches dedicated to solve this issue (Washha, 2018). They can be categorized as follows: Honeypot Approaches and machine learning approaches.

The Honeypot approaches require an intervention from the administrators of the systems. It is about an information system resource that can monitor social spammers behaviour through logging their information such as the information of accounts and any available content.

Regarding the machine learning methods, there are three levels of spam detection models:

Tweet-Level Detection which aims to predict the class label of tweet whether it is a spam or non-spam.

Account-Level Detection which focuses on deeply analysing the user's profile in order to predict the user of the account whether spammer or not.

Campaign-Level Detection which is interested in the examination of a group of accounts to judge whether it is a spam campaign or not.

Although all the cited methods are interesting, they focused only on how to add links to the network when an entity disappears in the case of the link prediction problem and on detecting spam and spammers without taking into consideration that the link can be spammed in the case of spammer detection problem.

In order to remedy this problem, we introduce in this chapter our third contribution (Ben Dhaou et al., 2019) which consists on detecting spammed links in social networks. Indeed, the proposed method consists on modelling the belief that a link is perceived as spammed by taking into account the prior information of the nodes, the links and the messages that pass through them.

To evaluate the proposed approach, the noise is added first to the messages, then to both links and messages in order to distinguish the spammed links in the network. Second, few spammed links of the network are selected and the proposed model is observed in order to determine if it manages to detect them.

This chapter is structured as follows. In section 4.2, the proposed method is in-

troduced. Then, we present in section 4.3 the obtained results. Finally, section 4.4 concludes the chapter.

4.2 Spammed Links Detection based on Nodes, Links and Messages Attributes

In social networks, several types of messages are exchanged among which, we find advertising messages, or other that we do not want to receive. In order to sort out our contacts in social networks, it became important to know which links are spammed.

In this work, a spammed link is considered as any link whose class has been modified because of the messages that pass through it in all the iterations. In one iteration, the mass function of the link is updated and it will be the input of the next iteration.

In order to model our idea, a belief graph $G = \{V^b; E^b\}$ is used, with: V^b a set of nodes and E^b a set of edges.

In this paper, three frames of discernment are considered for nodes, links and messages:

- $\Omega_N = \{\omega_{n_1}, \ldots, \omega_{n_N}\}$ for the set of nodes.
- $\Omega_L = \{\omega_{l_1}, \ldots, \omega_{l_L}\}$ for the set of links.
- $\Omega_M = \{\omega_{m_1}, \ldots, \omega_{m_M}\}$ for the set of messages.

Figure 4.1 presents the considered evidential graph in this work: A mass function is associated to each node, the link connecting them as well as the message transiting on it. In addition, a network with N communities is considered. Each community has a definite type that has been defined according to the type of links that make it up.

Figure 4.2 presents the proposed approach to detect spammed links. In order to integrate the belief on the links and on the messages, we first make a vacuous extension on $\Omega_L \times \Omega_M$ for each mass of the message of M^b and for each mass of the edge of E^b. Therefore, we obtain on each message M_i^b a mass: $m_i^{\Omega_L \times \Omega_M}$ and on each edge $E_{ij} = (V_i^b, V_j^b)$ between the nodes V_i^b and V_j^b a mass: $m_{ij}^{\Omega_L \times \Omega_M}$.

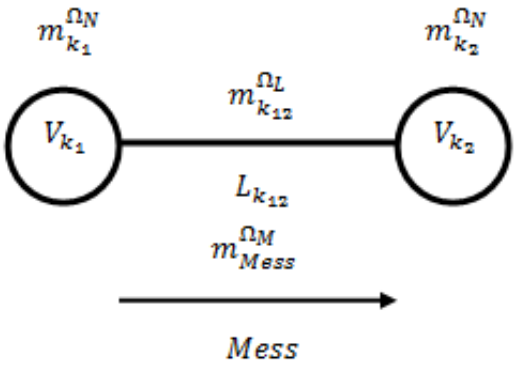

Figure 4.1: An Evidential Graph.

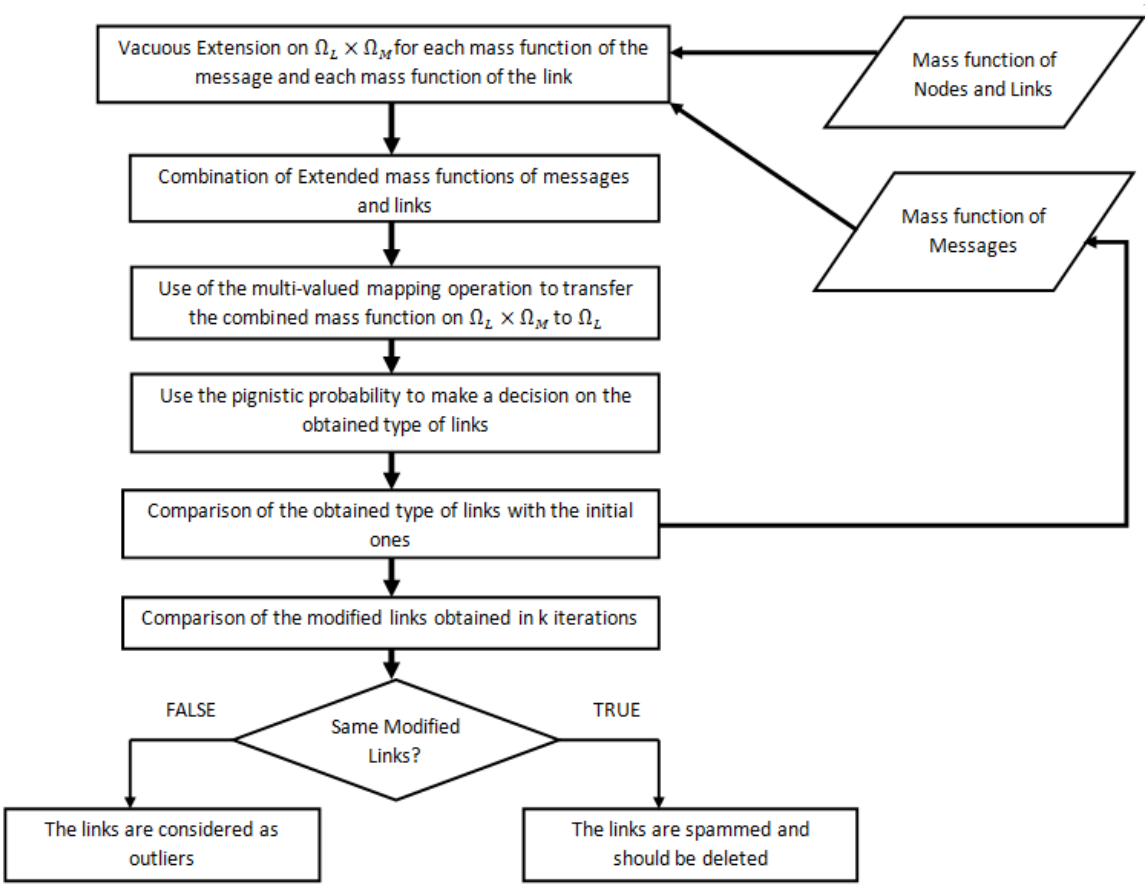

Figure 4.2: Process of the belief approach

Then, the extended mass functions are combined using the combination rule

of Dempster:

$$m^{\Omega_L \times \Omega_M} = m_{E_{ij}}^{\Omega_L \uparrow \Omega_L \times \Omega_M} \oplus m_{M_i}^{\Omega_M \uparrow \Omega_L \times \Omega_M} \tag{4.1}$$

The multi-valued operation is used to transfer the combined mass functions on $\Omega_L \times \Omega_M$ to Ω_L. In fact, a multi-valued mapping Γ describes a mapping function:

$$\Gamma : \Omega_L \times \Omega_M \rightarrow \Omega_L \tag{4.2}$$

These equations can be calculated by using the formula:

$$\Gamma : m_{\Gamma}^{\Omega_L}(B_j) = \sum_{\Gamma(e_i)=B_j} m^{\Omega_L \times \Omega_M}(e_i) \tag{4.3}$$

with $e_i \in \Omega_L \times \Omega_M$ and $B_j \subseteq \Omega_L$.

Thereafter, the pignistic probability is used in order to make a decision on the obtained type of links. This operation is used to make a comparison with the initial classes of links.

Since the proposed algorithm is iterative, we decide that a link is spammed and must be removed if its class changes at all iterations.

In this work, we have not developed a strategy for dealing with outliers. This will be the subject of future work. Indeed, for links that appear in some iterations but not all, we can set a threshold that represents the number of appearance of a spammed link and if it is greater than the threshold then this link could be considered as spammed.

4.3 Experimentations

In this section, we present the results obtained after applying the proposed algorithm. In this work, 3 LFR networks composed of 99 nodes with 468 links, 200 nodes with 818 links and 300 nodes with 1227 links are used. All the networks have 3 communities. In addition, 3 frames of discernment are considered:

- $\Omega_N = \{C_1, C_2, C_3\}$ for the nodes.
- $\Omega_L = \{Friendly, Family, Professional\}$ for the links.
- $\Omega_M = \{PNC, PC, INC, IC\}$ for the messages,

Γ	Friendly	Family	Professional
PNC	×	×	
PC	×	×	
INC			×
IC			×

Table 4.1: Definition of function Γ given the correspondences between $\Omega_L \times \Omega_M$ and Ω_L

with *PNC* for *PNC* for Personal Not Commercial, *PC* for Personal Commercial, *INC* for Impersonal Not Commercial and *IC* for Impersonal Commercial.

In this experiment, few LFR networks with three communities are considered. We assume that the first community is of type "friendly", the second of type "family" and the third is of type "professional". The type of community is defined from the types of links that make up the majority.

We start by generating the mass functions on nodes and links according to the structure of the network.

- For each node of the network, two focal elements are generated, one on the type of the node and the second on Ω_N by placing the largest value on the node type.
- For network links, two focal elements are also generated, one on the type of link and the second on Ω_L by assigning the largest value to the link type.

Then, the mass functions on the messages are generated depending on the link type. For each message which transits on the network, 2 focal elements are generated, one on the corresponding type of the message and the second on Ω_M.

Unlike the nodes and links of the network, we generate new mass functions on the messages at each iteration.

We use the passage function Γ defined in table 4.1 to transfer the mass functions from $\Omega_L \times \Omega_M$ to Ω_L.

In order to validate the proposed approach, two types of experiments are performed:

- The first type: adding noise on the messages only, then adding noise on the messages in addition of the links.
- The second: pre-selection of a number of spammed links and see if the proposed approach detects them.

In this work, we consider a noisy element (*i.e.* a link or a message) as an element whose mass function or probability has been modified and generated randomly. For the first part of the experiment, the noise is varied as follows:

- Case of noisy messages only: 20%, 40%, 50% and 70% of messages from each community were noisy.
- Case of noisy messages and noisy links
  - 20% of messages from each community were noisy and 20% of network links were noisy.
  - 40% of messages from each community were noisy 40% of network links were noisy.
  - 50% of messages from each community were noisy 50% of network links were noisy.
  - 70% of messages from each community were noisy 70% of network links were noisy.

4.3.1 Baseline

In order to show the efficiency of the proposed method, an algorithm that uses the same principle was performed in a probabilistic version.

Figure 4.3 shows the considered probabilistic graph: a vector of probabilities is assigned to each node, the link connecting them as well as the message transiting on it.

The probabilistic method consists of projecting the probabilities of links and messages on the Cartesian frame. Then, they are combined using the average. This will make it possible to know the type of the link according to the messages which transit on it.

Figure 4.4 presents the process steps explained before.

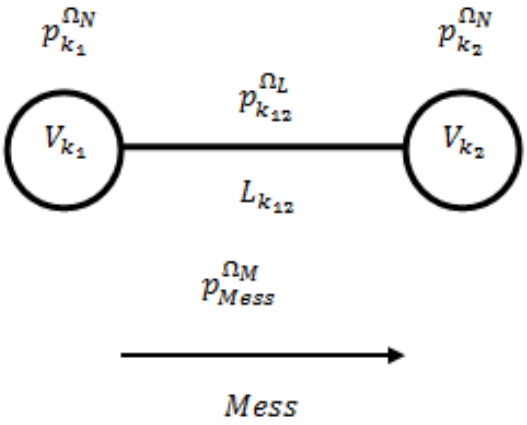

Figure 4.3: A Probabilistic Graph.

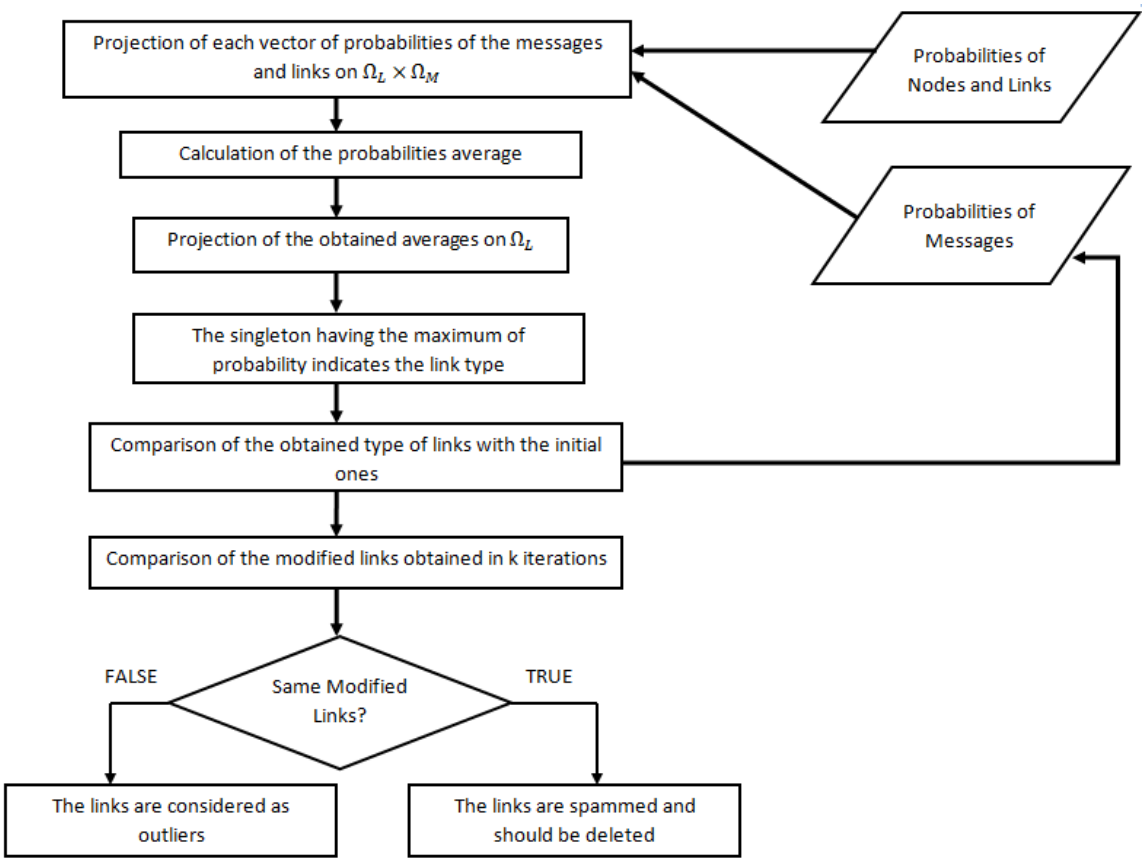

Figure 4.4: Process of the probabilistic approach

Extension of probabilities in the Cartesian product

Let the frames of links and messages in a general case:

- $\Omega_L = \{\omega_{l_1}, \omega_{l_2}, ..., \omega_{l_L}\}$
- $\Omega_M = \{\omega_{m_1}, \omega_{m_2}, ..., \omega_{m_M}\}$

The Cartesian frame is given by:
$\Omega_L \times \Omega_M = \{(\omega_{l_1}, \omega_{m_1}), (\omega_{l_1}, \omega_{m_2}), ..., (\omega_{l_L}, \omega_{m_M})\}$
Let 2 vectors of probabilities:
$P_L = (P_{\omega_{l_1}}, P_{\omega_{l_2}}, ..., P_{\omega_{l_L}})$ and $P_M = (P_{\omega_{m_1}}, P_{\omega_{m_2}}, ..., P_{\omega_{m_M}})$.

Given that the frames of the links and messages are independent, we need to project both probability vectors on the Cartesian frame $\Omega_L \times \Omega_M$ in order to combine them.

The fact that the theory of probabilities cannot model ignorance forces us to use an equi-probability when moving from one frame of discernment of links or messages to the Cartesian frame.

Hence for a given probability $P_L = (\omega_{l_i}, i = 1, ..., L)$, we consider the equi-probability on Ω_M to model the ignorance. The result is affected to each pair of Cartesian frame containing ω_{l_i}. For example:

$$\begin{aligned} P_L^{\Omega_L \times \Omega_M}(\omega_{l_1}, \omega_{m_1}) &= \frac{P_{\omega_{l_1}}}{|\Omega_M|}, \ldots, P^{\Omega_L \times \Omega_M}(\omega_{l_1}, \omega_{m_M}) = \frac{P_{\omega_{l_1}}}{|\Omega_M|}, \\ P_L^{\Omega_L \times \Omega_M}(\omega_{l_2}, \omega_{m_1}) &= \frac{P_{\omega_{l_2}}}{|\Omega_M|}, \ldots, P^{\Omega_L \times \Omega_M}(\omega_{l_2}, \omega_{m_M}) = \frac{P_{\omega_{l_2}}}{|\Omega_M|}, \\ &\ldots \end{aligned}$$

By the same process, in order to consider the probability $P_M = (\omega_{m_j}, j = 1, ..., M)$ in the Cartesian space $\Omega_L \times \Omega_M$, we consider the equi-probability on Ω_L to model the ignorance. For example:

$$\begin{aligned} P_M^{\Omega_L \times \Omega_M}(\omega_{l_1}, \omega_{m_1}) &= \frac{P_{\omega_{m_1}}}{|\Omega_L|}, \ldots, P^{\Omega_1 \times \Omega_M}(\omega_{l_L}, \omega_{m_1}) = \frac{P_{\omega_{m_1}}}{|\Omega_L|}, \\ P_M^{\Omega_L \times \Omega_M}(\omega_{l_1}, \omega_{m_2}) &= \frac{P_{\omega_{m_2}}}{|\Omega_L|}, \ldots, P^{\Omega_2 \times \Omega_M}(\omega_{l_L}, \omega_{m_2}) = \frac{P_{\omega_{m_2}}}{|\Omega_L|}, \\ &\ldots \end{aligned}$$

Calculation of the average of the probabilities

Once the probabilities of the links and messages are projected on the Cartesian frame, we proceed then to the combination of both vectors of probabilities using the average.

In this work, we chose to use the average because it has a compromise behaviour. Indeed, if the data contain estimation errors, the calculation of the average makes it possible to reduce this rate of error. For example:

$$\frac{P_L^{\Omega_L \times \Omega_M}(\omega_{l_1}, \omega_{m_1}) + P_M^{\Omega_L \times \Omega_M}(\omega_{l_1}, \omega_{m_1})}{2},$$
$$\frac{P_L^{\Omega_L \times \Omega_M}(\omega_{l_2}, \omega_{m_2}) + P_M^{\Omega_L \times \Omega_M}(\omega_{l_2}, \omega_{m_2})}{2},$$
$$\dots$$

Projection of obtained averages on the frame of links

In order to return to the frame of the links, we proceed by summing the average probabilities of the hypotheses that are related to each type of link $(\omega_{l_i}, \omega_{m_j})$, $i = 1, \dots, L$; $j = 1, \dots, M$.

Decision making

From each probability vector relative to each link, we determine the current type of the given link $max(\omega_{l_i}), i = 1, ..., L$. Hence, we compare the obtained type with the initial one and decide if the link is spammed or not.

4.3.2 Case of noisy messages only

In this section, we present the results obtained after adding 20%, 40%, 50% and 70% of noisy messages in each community. The histograms given on Figures 4.5, 4.6, 4.7 and 4.8 show respectively the number of spammed links that appeared after 5, 10, 15 and 20 iterations.

It is noticed that the more the percentage of the noisy messages increases the more the number of spammed links increases likewise. It is also remarked that in the case of the baseline a larger number of links would be removed compared to the belief approach. This could cause disconnection of the network.

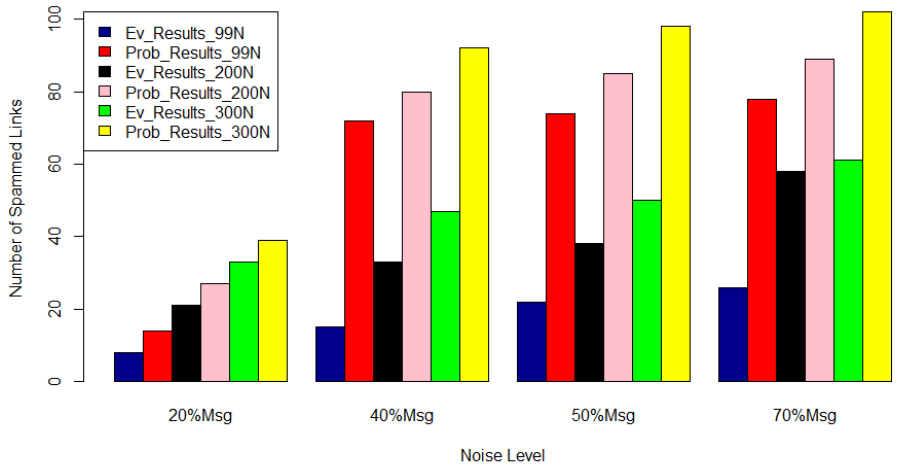

Figure 4.5: Spammed links after 5 iterations: case of noisy messages only.

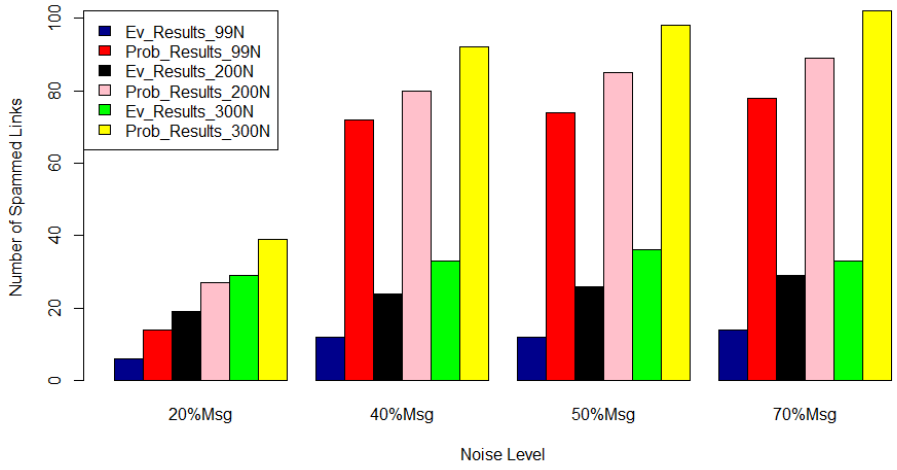

Figure 4.6: Spammed Links after 10 iterations: case of noisy messages only.

4.3.3 Case of noisy messages and noisy links

In this section, we present the results after adding 20%, 40%, 50% and 70% of the noisy messages in each community in addition of 20%, 40%, 50% and 70% of the noisy links picked randomly from each network.

The histograms given in Figures 4.9, 4.10, 4.11 and 4.12 show respectively

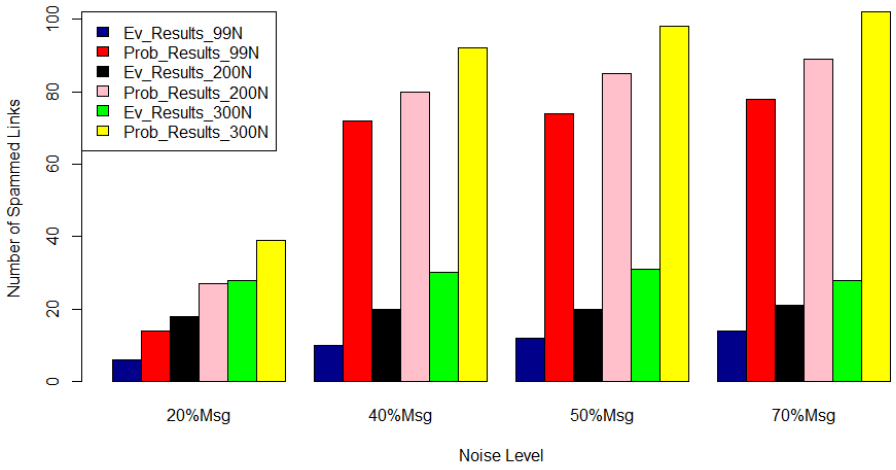

Figure 4.7: Spammed links after 15 iterations: case of noisy messages only.

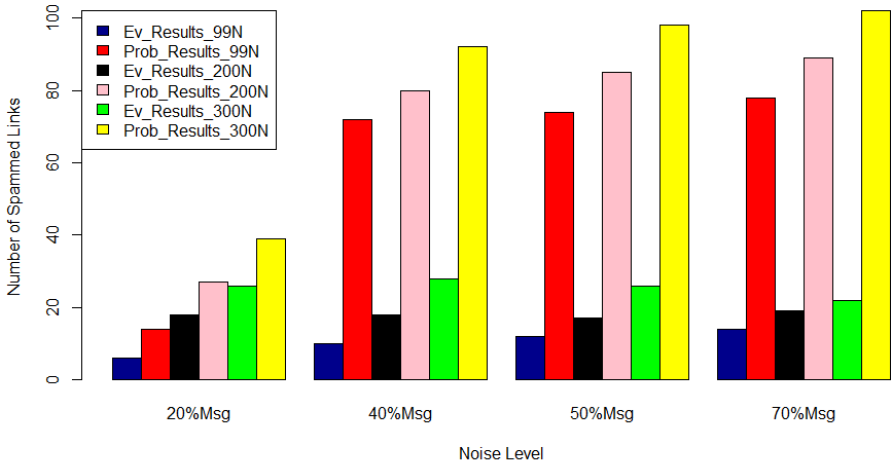

Figure 4.8: Spammed links after 20 iterations: case of noisy messages only.

the number of spammed links that appeared after 5, 10, 15 and 20 iterations while varying noise.

We note that the baseline begets the removal of a large number of network links. As a result, the network is no longer connected. For example, in the case of 70% noisy messages and 70% noisy links, it detects 213 links which represents about 45.5% of the total links of the network composed of 99 nodes. The proposed

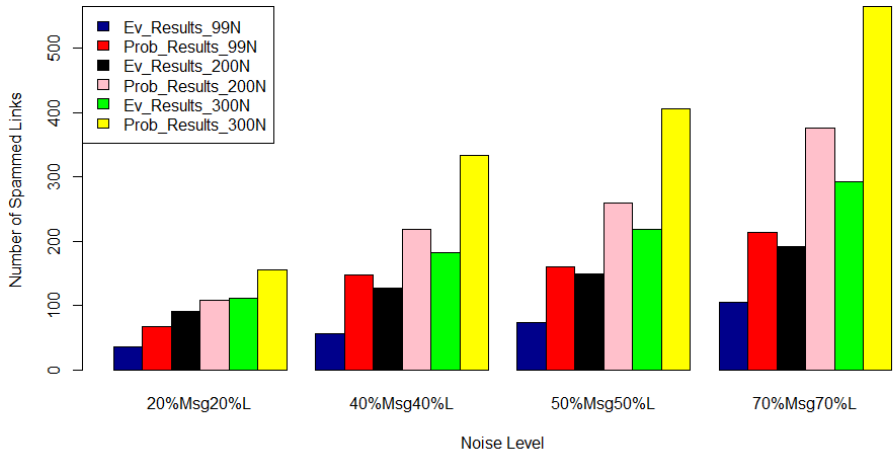

Figure 4.9: Spammed links after 5 iterations: case of noisy messages and links.

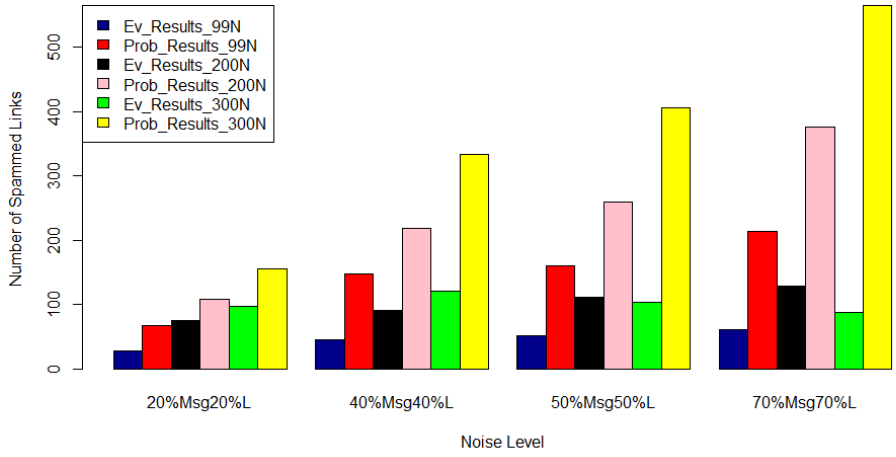

Figure 4.10: Spammed links after 10 iterations: case of noisy messages and links.

approach provides better results than the baseline due to the fact that the theory of belief functions manages better ignorance and conflict.

4.3.4 Detection of Spammed Links

In this section, we present the obtained accuracy results after 10 iterations.

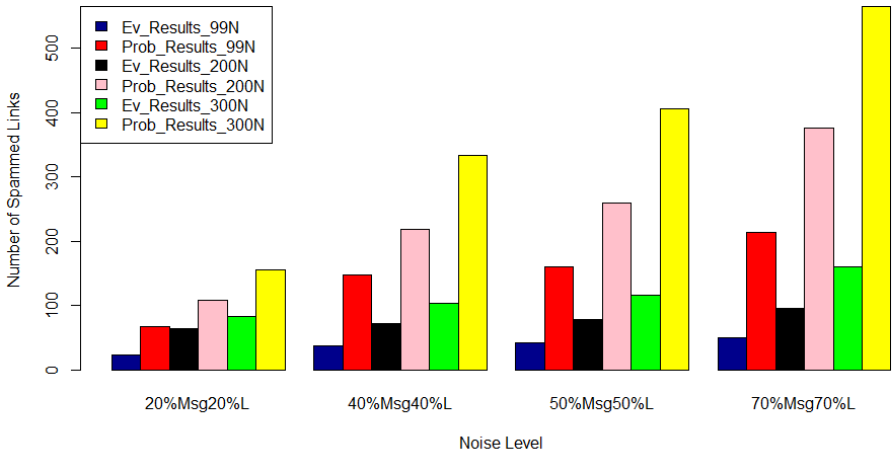

Figure 4.11: Spammed links after 15 iterations: case of noisy messages and links.

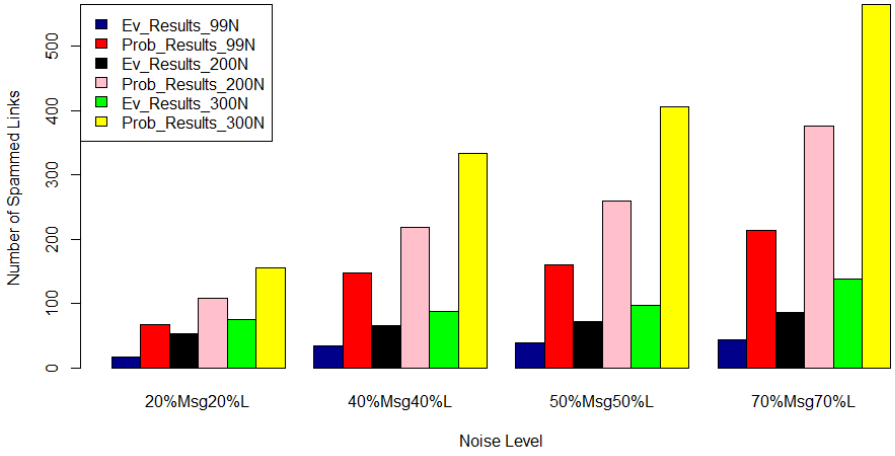

Figure 4.12: Spammed links after 20 iterations: case of noisy messages and links.

The goal of this experiment is to test if our model manages to detect the known spammed links. The generated mass functions on the messages are not compatible with the spammed links classes. We consider a LFR network composed of 99 nodes and 10 spammed links.

The obtained results given by the proposed approach, the baseline and the k-nn algorithm are compared.

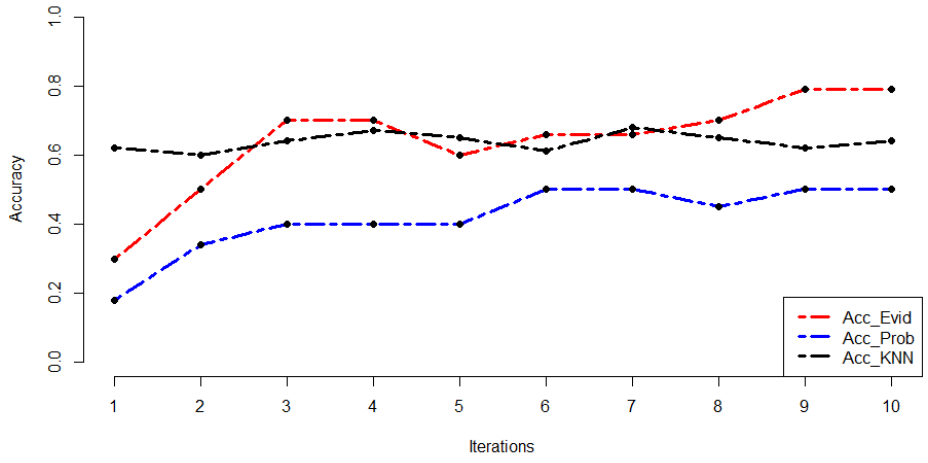

Figure 4.13: Accuracy Results: Case of $PNCUPC$.

The k-nearest neighbour (k-nn) (Altman, 1992) is a supervised learning method. Its principle is as follows: An object is classified by a majority vote of its neighbours, with the object being assigned to the class most common among its k nearest neighbours.

It should be noted that in Figures 4.13, 4.14 and 4.15, the accuracy values given by the k-nn oscillate between 0.6 and 0.69. This is because the k-nn requires learning data in contrary to the proposed approach and the baseline. In the following, the results of 3 cases are presented:

Generation of 10 **messages of type** $PNCUPC$ The spammed links are of type "professional". Hence, 10 incompatible messages of type "PNC U PC" are generated. The curves in figure 4.13 show that for both evidential and probabilistic approaches, only few spammed links were detected at the first iteration. However, the evidential accuracy is higher than the probabilistic one. For the case of the k-nn algorithm, we notice that it has better accuracy results at the first iterations. Nevertheless, at the tenth iteration, it is noticed that the evidential accuracy becomes equal to 79%. So, we can conclude that our model is able to detect correctly more spammed links than the baseline and the k-nn algorithm.

Generation of 10 **messages of type** PNC, PC **and** $PNCUPC$ We generate:

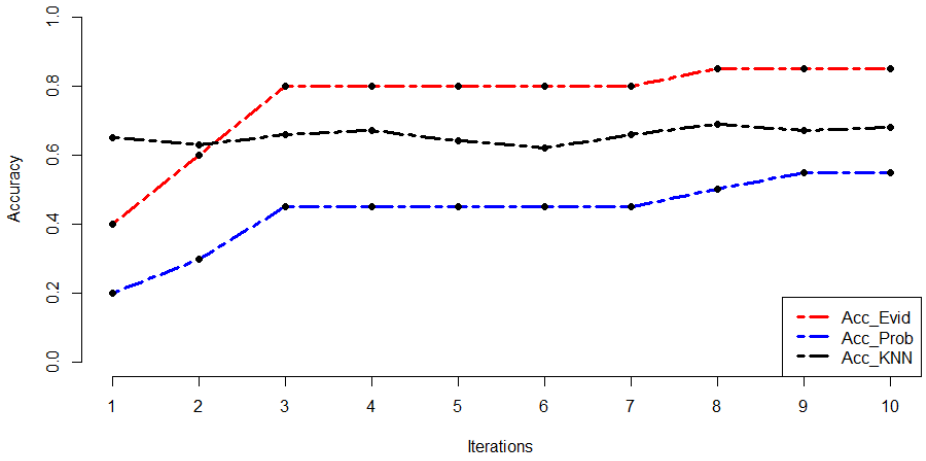

Figure 4.14: Accuracy Results: Case of PNC, PC, and $PNCUPC$.

- 3 messages of type PNC,
- 3 messages of type PC
- and 4 messages of type $PNCUPC$.

In Figure 4.14 we can note a clear improvement of detection of spammed links at the tenth iteration. Indeed, the evidential accuracy results given by the proposed approach is equal to 85%.

Generation of 10 **messages of type** $PNCUPC$ **and random** We generate:

- 6 random messages
- 4 messages of type $PNCUPC$.

We specify that in the case of random message, the focal element can be everywhere except on the empty set in the case of the proposed model.

Figure 4.15 shows that even when we have a portion of random messages generated on spammed links, our model always gives the best results of accuracy at the tenth iteration and even before. These results can be explained by the fact that the theory of belief functions offers a strong tool to handle the imperfection of the information.

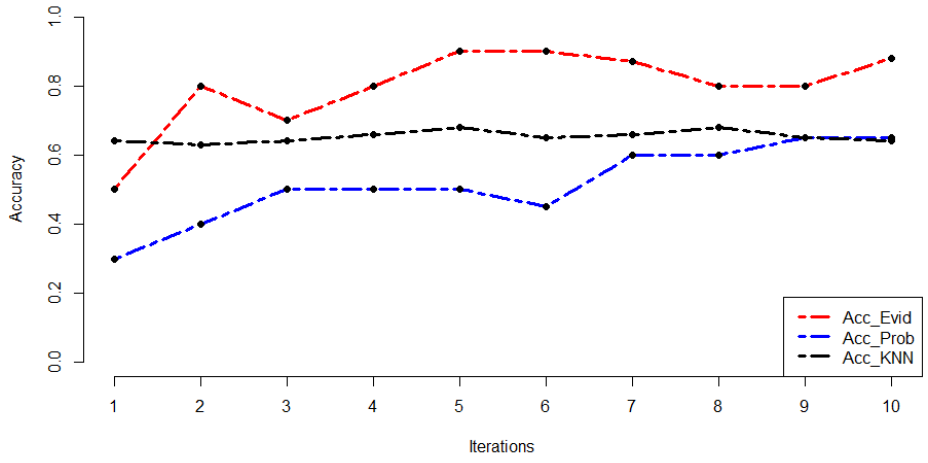

Figure 4.15: Accuracy Results: Case of random and *PNCUPC* messages.

Evaluation of the algorithm in terms of precision and recall

In this section, we present the obtained precision and recall results of the proposed approach, the baseline and the *k*-nn algorithm.

The effectiveness of an information retrieval technique is measured using two separate measurements: the precision and recall. Precision is the fraction of relevant instances among the retrieved instances while recall is the fraction of relevant instances that have been retrieved over the total amount of relevant instances.

In what follows, we present the obtained average precision and average recall for probabilistic and evidential approaches in addition of the *k*-nn algorithm in the case of an LFR network composed of 200 and 400 nodes.

Case of LFR network 200 **Nodes** We start by spamming 60 links of network as follows:

- 20 links of type “professional”
- 20 links of type “friendly”
- 20 links of type “family”.

For each type of links, 20 incompatible message were generated:

- For the case of the "professional" link, we generate messages of type "PNC", "PC" and "PNC U PC".
- For the case of the "friendly" and "family" links, we generate messages of type "IC", "INC" and "IC U INC".

The curves in Figure 4.16 show a comparison of the obtained results in terms of precision and recall measures in the case of the proposed approach, the baseline and the *k*-nn. We represent the obtained values at the first and tenth iteration.

The first point of each curve represents the result obtained at the first iteration and the second point represents the result obtained at the tenth iteration.

We note that the results given by the *k*-nn at the first and tenth iterations are close. This is due to the fact that this algorithm requires learning data unlike the evidential and probabilistic methods. Therefore, the methods do not compare the same thing.

We notice also that the proposed algorithm gives better results than the baseline and the *k*-nn algorithm. To sum up, there is a learning difference between the proposed approach, the baseline and the *k*-nn algorithm. Indeed, the proposed algorithms have no prior knowledge and they are used to understand and explore the data. However, the *k*-nn algorithm is based on training set and used to classify future data.

Case of LFR network 400 **nodes** This experiment was performed on an LFR network composed of 400 nodes, 1864 links and 3 communities. In addition, 600 links were spammed. The obtained results at the first and tenth iteration are presented.

Figure 4.17 shows that the proposed approach gives better results in terms of precision and recall compared to the baseline and the *k*-nn algorithm. We remind that the closeness of the results given by the *k*-nn at the first and tenth is due to the fact that this algorithm requires learning data unlike the evidential and probabilistic methods.

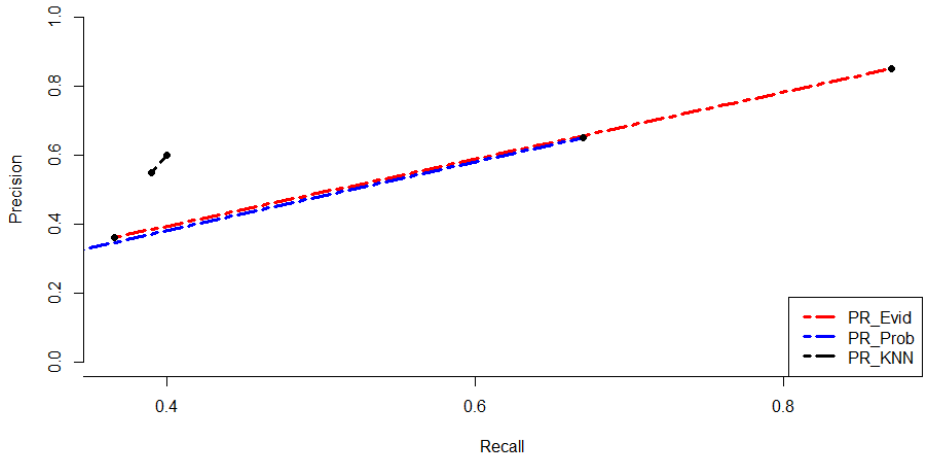

Figure 4.16: Precision and Recall Results at first and tenth iterations.

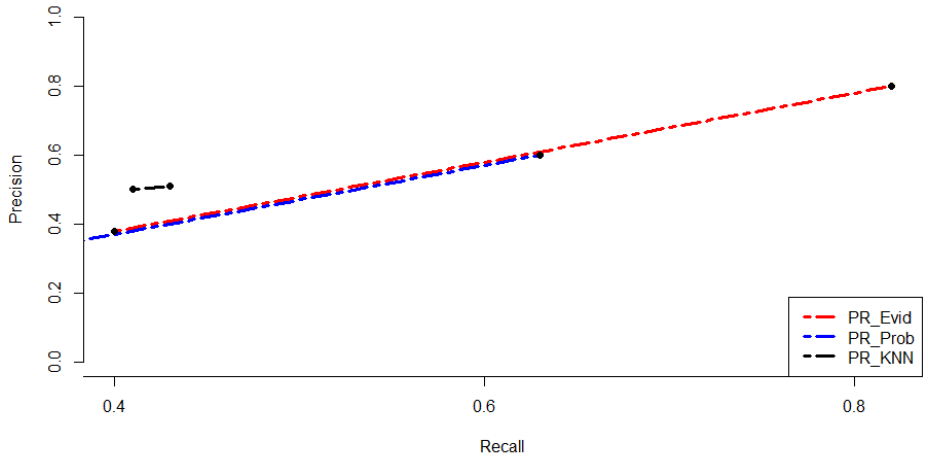

Figure 4.17: Comparison of the precision/recall results at the first and the tenth iteration.

4.4 Conclusion

Throughout this chapter, the third contribution which consists on detecting spammed links using the information of the nodes, links and messages was explained.

In order to test the proposed approach, two types of experimentations were performed:

First, the noise is added on the messages only, and then we added noise on both messages and links.

Second, we selected spammed links and observed if the proposed model manages to detect them.

The noise was varied on 4 LFR networks composed of 3 communities and having different sizes. The obtained results were compared with those given by the probabilistic approach and the k-nn algorithm.

Experiments have shown that, as expected, the number of spammed links increases with the noise level. Indeed, the higher the noise rate increases, the more the attributes become inconsistent with the network structure as well as the type of links. In addition, the results showed that the belief approach is better than the probabilistic one since the latter delete almost half of the network links.

Furthermore, the accuracy, precision and recall results prove that our model is able to detect the majority of spammed links and gives better results than the considered baseline and the k-nn algorithm.

Conclusion and Future Works

Conclusion

A social network refers to a set of individuals who are related and interact by exchanging content. A social network can be represented by a graph $G(V, E)$ where V represents the set of vertices (persons, institutions, etc) and E the set of edges (relationships).

The entities of a social network can be of different types: web pages, members of a social site, bank accounts, proteins, etc. As for the links, they can represent varied interactions between these entities: friendship links on Facebook, follower on Twitter, hyper-links on the web, etc.

In a social network we can find many communities. We recall that the community represents a group of people who have special ties because they have particular affinities, or have similar characteristics, or share interests, etc.

From a point of view of graph theory, a community is considered as a subgraph composing of a set of nodes that are strongly linked to each other, and loosely related to the nodes located outside the community.

The role of community detection is to highlight those groups that have formed implicitly. We find many interest of the community detection task such that identifying profile types, carrying out targeted actions, better adjusting the recommendations, identifying influential actors, etc.

In the field of social network analysis, we manipulate information that is often imperfect. For a long time, it was considered that the probabilistic framework

was the only framework adapted to the representation and the manipulation of imperfect data. However, other theories of imprecise and uncertain management such as theories of possibilities and belief functions have emerged in order to reduce the imperfection of an information.

Keeping all this in mind, we focused on this thesis on community detection in an uncertain framework. We chose to use theory of belief functions for the advantages it offers such as modelling and management of imprecision and uncertainty. In addition, it offers a very good mathematical tools for the fusion of data provided by several sources.

In what follows, we summarise the contributions presented throughout this report.

In this thesis, we were first interested in studying the advantage of using evidential attributes in the detection of communities in social networks. Indeed, we have compared the clustering results of three types of uncertain attributes: numerical, probabilistic and evidential. We considered 2 scenarios in the experiments: the clustering of random generated data and the clustering of sorted one. In addition, we tested the proposed algorithm in the case of the presence of noisy information. The results of the Normalized Mutual Information (NMI) show that nodes with evidential attributes are better classified than nodes with numerical or probabilistic attributes.

After that, we proposed a method which allows the classification based on the structure of the network as well as on the attributes of the nodes and links. Indeed, the aim of this algorithm is to correct the noise added to the information of the network. In order to validate our proposed approach, we considered 3 cases: the nodes only are noisy, the links only are noisy and finally, both nodes and links are noisy. We also tested our algorithm by varying the different LFR parameters such as the size of network, the number of communities and the mixing parameter μ. We compared our results with those given by the baseline. It has been shown that the proposed method gives the best results. This can be explained by the fact that the theory of belief functions manages ignorance as well as conflict.

The third contribution consists of detecting spammed links in social networks. Indeed, using the information of the nodes, links and messages, we model the belief that a link is perceived as spammed or not. In order to evaluate the proposed contribution, we performed 2 sets of experiments: first, we added noise to the messages only and then to both links and messages. We compared the obtained

results with those of the baseline and the k-nn algorithm. The proposed algorithm gives the best detection results. This is because the theory of belief functions is a strong tool for managing uncertainty, ignorance, imprecision as well as conflict.

To sum up, our work on modelling interactions between nodes in a credibilist social network aims to detect communities in the presence of imperfect information based on both structure of the network and attributes associated to the entities composing this latter using the theory of belief functions. We deal with imperfect information because in social networks, the information related to the nodes, links and messages can be often imprecise, uncertain or ambiguous due to the heterogeneous nature of sources.

Future Works

In this section, we try to develop some possible future works based on the current work that we have already done in this thesis. We are mostly interested in several directions as below.

First, we intend to correct noisy informations in the case of overlapping communities. The idea is to calculate the distances between the triplets of the network and the coherent ones. However, the initial generated mass functions on the nodes will have two focal elements: one on the union of the communities to which the nodes belong and the second one on Ω_N. As for the categorical mass functions of the coherent triplets, they will have a unique focal element on the union of the communities to which the nodes belong according to the structure of the network.

Second, the improvement of the running time of the algorithm proposed in the second contribution will be considered. Indeed, we intend to reduce its running time. In fact, there are several strategies that can reduce complexity such as representing only the focal elements or grouping them together if their values are negligible (Martin, 2009).

Third, for the case of dealing with outliers in the spammed links detection, we aim to fix a threshold that represents the minimum number of occurrences for a link to be considered spammed. We remind that an outlier is a link that its initial class can be modified but not in all iterations.

Scaling Up is a problem that will also be considered. In fact, we intend to test

our proposed contributions on real and large social networks such as Facebook, LinkedIn and Twitter. However, when considering scalability, it should be kept in mind that the algorithms become NP-complete.

Among the problems related to community detection, the most general and difficult problem remains the detection of all relevant groups in a large real network i.e the detection of overlapping communities. Indeed, although solutions to this problem have been proposed (Xie et al., 2013) such as clique percolation, line graph and link partitioning, local expansion and optimization, fuzzy detection as well as agent based and dynamical algorithms, no solution is unanimous and is not entirely satisfactory. Therefore, we intend to develop a method based on the structure of the network as well as the nodes and links attributes in order to detect overlapping communities.

The use of deep learning for the detection of communities represents a very interesting perspective to discover. In fact, several works of the literature such as (L. Yang et al., 2016) were focused on identifying community structure using deep learning. As a future work, we intend to combine the deep learning with the theory of belief functions in order to produce effective community detection models.

Another interesting perspective is to test the proposed methods in this thesis on biological networks. Indeed, instead of considering the vertices as a social entities, they will be associated to biological entities such as proteins, genes, metabolites, etc. As for the links, instead of representing friendship, familial or professional relationship, they will represent transformation of molecules into other molecules such as chemical reactions, expression of a gene or formation of a complex, etc. Among the things studied in biological networks, there is the prediction of adverse effects of drugs (M. Liu et al., 2012). From there, the third approach proposed in this thesis could be used to detect these undesirable effects instead of detecting spammed links.

Appendix A

LFR Parameters

In this work, we used the LFR parameters presented in table A.1 for the generation of our networks. In the follwoing, we remind the meaning of each parameter:

- N represents the number of nodes,
- k the average degree,
- *maxk* the maximum degree,
- *mu* the mixing parameter,
- $t1$ the minus exponent for the degree sequence,
- $t2$ the minus exponent for the community size distribution,
- *minC* the minimum for the community size,
- *maxC* the maximum for the community size,
- *on* the number of overlapping nodes,
- *om* the number of memberships of the overlapping nodes
- and C the average clustering coefficient.

N	k	maxk	mu	t1	t2	minC	maxC	on	om	C
99	5	10	0.3	2	1	33	33	0	0	0.55
200	5	10	0.3	2	1	66	67	0	0	0.55
200	5	10	0.3	2	1	50	50	0	0	0.55
200	5	10	0.3	2	1	40	40	0	0	0.55
200	5	10	0.3	2	1	33	33	0	0	0.55
300	5	10	0.3	2	1	100	100	0	0	0.55
400	5	10	0.3	2	1	132	135	0	0	0.55
50	5	10	0.3	2	1	15	17	0	0	0.55
200	5	10	0.1	2	1	66	67	0	0	0.55
200	5	10	0.5	2	1	66	67	0	0	0.55
200	5	10	0.7	2	1	66	67	0	0	0.55
200	5	10	0.9	2	1	66	67	0	0	0.55

Table A.1: Parameters of LFR

Appendix B

Results Before Adding Noise

In this Appendix, we show the obtained results of the experiments performed on 3 other LFR networks.

B.1 LFR Network: 50 Nodes +3 Communities

First Scenario We present below the average values of NMI for 100 runs of random generated attributes in the LFR network composed of 50 nodes and 3 communities.

We notice that the average evidential NMI is the highest value comparing to the

	NMI-Average	Interval of Confidence
Numerical	0.748	$[0.606, 0.89]$
Probabilistic	0.707	$[0.644, 0.853]$
Evidential	1	$[1, 1]$

Table B.1: NMI Averages et Intervals of Confidence- Case LFR 50 Nodes: First Scenario.

	NMI-Average	Interval of Confidence
Probabilistic	0.839	[0.795, 0.882]
Evidential	1	[1, 1]

Table B.2: NMI Averages et Intervals of Confidence- Case LFR 50 Nodes: Second Scenario.

probabilistic and the numerical ones. The algorithm K-medoids is able to affect all the nodes in their right cluster based on their evidential attributes.

Second Scenario We proceed to sort the matrix of generated attributes by putting the highest generated value on C_1, C_2 or C_3, depending on the belonging of the node. Then, we compute the average values of NMI for 100 executions.

We notice that the clustering with evidential attributes gives an average NMI value equal to 1 comparing to the probabilistic ones. We also notice that the K-medoids was not able to classify all the nodes in their right clusters based on their probabilistic attributes.

B.2 LFR Network: 99 Nodes + 3 Communities

First Scenario In this section, we show the results of the NMI computation of the random generated attributes. We present below the results of the average values of NMI for 100 runs of random attributes generation in the case of an LFR network composed of 99 nodes and 3 communities.

The results show that the evidential generated attributes give better results than the probabilistic and the numerical ones. In fact, we obtained a value of the NMI average equal to 1 which means that the clustering algorithm K-medoids is able to classify the nodes according to their evidential attributes in the right cluster.

	NMI-Average	Interval of Confidence
Numerical	0.671	[0.596, 0.745]
Probabilistic	0.686	[0.550, 0.821]
Evidential	1	[1, 1]

Table B.3: NMI Averages et Intervals of Confidence- LFR 99 Nodes: First Scenario.

	NMI-Average	Interval of Confidence
Probabilistic	0.860	[0.821, 0.9]
Evidential	1	[1, 1]

Table B.4: NMI Averages et Intervals of Confidence- LFR 99 Nodes: Second Scenario.

Second Scenario We executed the generation of the attributes several time and we sorted the matrix of attributes. We obtain the results of the average values of NMI for 100 executions below:

The results show that the evidential version gives an average NMI value equal to 1, which means that each node was detected in the right cluster. We notice that after sorting the probabilistic attributes, the K-medoids was able to affect only 86% of the nodes in their right clusters.

B.3 LFR Network: 200 Nodes + 3 Communities

First Scenario In this part, we present the obtained results of the NMI average values in the case of an LFR network composed of 200 nodes and 3 communities for 100 runs of random generated attributes.

	NMI-Average	Interval of Confidence
Numerical	0.700	[0.665, 0.735]
Probabilistic	0.776	[0.737, 0.815]
Evidential	1	[1, 1]

Table B.5: NMI Averages et Intervals of Confidence- LFR 200 Nodes: First Scenario.

	NMI-Average	Interval of Confidence
Probabilistic	0.839	[0.809, 0.87]
Evidential	1	[1, 1]

Table B.6: NMI Averages et Intervals of Confidence- LFR 200 Nodes: Second Scenario.

The results show that the clustering based on the generated evidential attributes gives better results than the probabilistic and the numerical ones. In fact, all the nodes were affected to their correct clusters and this is confirmed by the value of the NMI average which is equal to 1.

Second Scenario We performed the generation of the attributes 100 times and we sorted the matrix of attributes (We put the highest value on the attribute C_1, C_2 or C_3 depending of the belonging of the node to C_1, C_2 or C_3). We obtain the results of the average values of NMI for 100 below:

The results show that the evidential version gives an average NMI value equal to 1 comparing to the probabilistic one which means that all the nodes were classified in their right clusters.

References

Adar, E., & Re, C. (2007). Managing uncertainty in social networks. *IEEE Data Eng. Bull.*, *30*(2), 15–22.

Al Hasan, M., & Zaki, M. J. (2011). A survey of link prediction in social networks. In *Social network data analytics* (pp. 243–275). Springer.

Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, *46*(3), 175–185.

Amaral, L. A. N., Scala, A., Barthelemy, M., & Stanley, H. E. (2000). Classes of small-world networks. *Proceedings of the national academy of sciences*, *97*(21), 11149–11152.

Arora, P., Varshney, S., & Deepali, D. (2016). Analysis of k-means and k-medoids algorithm for big data. *Procedia Computer Science*, *78*, 507–512.

Barabási, A.-L., & Albert, R. (1999). Emergence of scaling in random networks. *science*, *286*(5439), 509–512.

Bavelas, A. (1950). Communication patterns in task-oriented groups. *The Journal of the Acoustical Society of America*, *22*(6), 725–730.

Ben Dhaou, S., Kharoune, M., Martin, A., & Ben Yaghlane, B. (2014). Belief approach for social networks. In *International conference on belief functions* (pp. 115–123).

Ben Dhaou, S., Kharoune, M., Martin, A., & Ben Yaghlane, B. (2018). An evidential method for correcting noisy information in social network. *Online Social Networks and Media Journal*, *7*, 30–44.

Ben Dhaou, S., Kharoune, M., Martin, A., & Ben Yaghlane, B. (2019). A belief approach for detecting spammed links in social networks. In *Proceedings of the 11th international conference on agents and artificial intelligence - volume 2: Icaart,* (p. 602-609).

Ben Dhaou, S., Zhou, K., Kharoune, M., Martin, A., & Ben Yaghlane, B. (2017). The advantage of evidential attributes in social networks. In *20th international conference on information fusion* (pp. 1–8).

Dempster, A. P. (1967). Upper and lower probabilities induced by a multivalued mapping. *The annals of mathematical statistics*, 325–339.

Denceux, T. (2006). The cautious rule of combination for belief functions and some extensions. In *9th international conference on information fusion* (pp. 1–8).

Denœux, T. (2008). A k-nearest neighbor classification rule based on dempster-shafer theory. *Classic works of the Dempster-Shafer theory of belief functions*, 737–760.

Dubois, D., & Prade, H. (1988). Representation and combination of uncertainty with belief functions and possibility measures. *Computational intelligence*, *4*(3), 244–264.

Easley, D., & Kleinberg, J. (2010). *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press.

Erdos, P., & Rényi, A. (1960). On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci*, *5*(1), 17–60.

Essaid, A., Martin, A., Smits, G., & Ben Yaghlane, B. (2014, December). A Distance-Based Decision in the Credal Level. In *International Conference on Artificial Intelligence and Symbolic Computation (AISC 2014)* (p. 147 - 156). Sevilla, Spain.

Fortunato, S. (2010). Community detection in graphs. *Physics reports*, *486*(3), 75–174.

Freeman, L. C. (1978). Centrality in social networks conceptual clarification. *Social networks*, *1*(3), 215–239.

Girvan, M., & Newman, M. E. (2002). Community structure in social and biological networks. *Proceedings of the national academy of sciences*, *99*(12), 7821–7826.

Guimerà, R., & Sales-Pardo, M. (2009). Missing and spurious interactions and the reconstruction of complex networks. *Proceedings of the National Academy of Sciences*, *106*(52), 22073–22078.

Holland, P. W., & Leinhardt, S. (1971). Transitivity in structural models of small groups. *Comparative group studies*, *2*(2), 107–124.

Hu, Y., Chen, H., Zhang, P., Li, M., Di, Z., & Fan, Y. (2008). Comparative definition of community and corresponding identifying algorithm. *Physical Review E*, *78*(2), 026121.

Hubert, L., & Arabie, P. (1985). Comparing partitions. *Journal of classification*,

2(1), 193–218.

Jousselme, A.-L., Grenier, D., & Bossé, É. (2001). A new distance between two bodies of evidence. *Information fusion*, *2*(2), 91–101.

Kaufman, L., & Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis* (Vol. 344). John Wiley & Sons.

Khan, A., Bonchi, F., Gionis, A., & Gullo, F. (2014). Fast reliability search in uncertain graphs. In *17th international conference on extending database technology (edbt)* (pp. 535–546).

Knops, Z. F., Maintz, J. A., Viergever, M. A., & Pluim, J. P. (2006). Normalized mutual information based registration using k-means clustering and shading correction. *Medical image analysis*, *10*(3), 432–439.

Lancichinetti, A., Fortunato, S., & Radicchi, F. (2008). Benchmark graphs for testing community detection algorithms. *Physical review E*, *78*(4), 046110.

Lee, H. (2011). *Context Reasoning Under Uncertainty Based On Evidential Fusion Networks In Home-based Care* (PhD Thesis). University of Texas.

Lee, K., Caverlee, J., & Webb, S. (2010). Uncovering social spammers: social honeypots+ machine learning. In *Proceedings of the 33rd international acm sigir conference on research and development in information retrieval* (pp. 435–442).

Leskovec, J., Backstrom, L., Kumar, R., & Tomkins, A. (2008). Microscopic evolution of social networks. In *Proceedings of the 14th acm sigkdd international conference on knowledge discovery and data mining* (pp. 462–470).

Leskovec, J., & Mcauley, J. J. (2012). Learning to discover social circles in ego networks. In *Advances in neural information processing systems* (pp. 539–547).

Liu, M., Matheny, M. E., Hu, Y., & Xu, H. (2012). Data mining methodologies for pharmacovigilance. *ACM SIGKDD Explorations Newsletter*, *14*(1), 35–42.

Liu, Z.-G., Pan, Q., Mercier, G., & Dezert, J. (2015). A new incomplete pattern classification method based on evidential reasoning. *IEEE transactions on cybernetics*, *45*(4), 635–646.

Lusseau, D. (2003). The emergent properties of a dolphin social network. *Proceedings of the Royal Society of London B: Biological Sciences*, *270*(Suppl 2), S186–S188.

Mallek, S., Boukhris, I., Elouedi, Z., & Lefevre, E. (2015). The link prediction problem under a belief function framework. In *International conference on tools with artificial intelligence (ictai)* (pp. 1013–1020).

Marsden, P. V. (1988). Homogeneity in confiding relations. *Social networks*, *10*(1), 57–76.

Martin, A. (2009). Implementing general belief function framework with a practical codification for low complexity. *Advances and applications of DSmT for Information Fusion-Collected works*, *3*, 217–273.

Martin, A., & Osswald, C. (2006). Human experts fusion for image classification. *Information and Security*, *20*, 122–143.

Martin, A., & Osswald, C. (2007). Toward a combination rule to deal with partial conflict and specificity in belief functions theory. In *10th international conference on information fusion* (pp. 1–8).

Martinez-Romo, J., & Araujo, L. (2013). Detecting malicious tweets in trending topics using a statistical analysis of language. *Expert Systems with Applications*, *40*(8), 2992–3000.

McPherson, M., Smith-Lovin, L., & Cook, J. M. (2001). Birds of a feather: Homophily in social networks. *Annual review of sociology*, *27*(1), 415–444.

Meilă, M. (2003). Comparing clusterings by the variation of information. In *Learning theory and kernel machines* (pp. 173–187). Springer.

Mika, P. (2004). Social networks and the semantic web. In *Proceedings of the 2004 ieee/wic/acm international conference on web intelligence* (pp. 285–291).

Moradabadi, B., & Meybodi, M. R. (2017). Link prediction in fuzzy social networks using distributed learning automata. *Applied Intelligence*, *47*(3), 837–849.

Newman, M. E. (2004). Fast algorithm for detecting community structure in networks. *Physical review E*, *69*(6), 066133.

Newman, M. E., & Girvan, M. (2004). Finding and evaluating community structure in networks. *Physical review E*, *69*(2), 026113.

Ng, R. T., & Han, J. (2002). Clarans: A method for clustering objects for spatial data mining. *IEEE transactions on knowledge and data engineering*, *14*(5), 1003–1016.

Parchas, P., Gullo, F., Papadias, D., & Bonchi, F. (2014). The pursuit of a good possible world: extracting representative instances of uncertain graphs. In *Proceedings of the 2014 acm sigmod international conference on management of data* (pp. 967–978).

Pons, P., & Latapy, M. (2005). Computing communities in large networks using random walks. In *International symposium on computer and information sciences* (pp. 284–293).

Prell, C. (2012). *Social network analysis: History, theory and methodology*. Sage.

Radicchi, F., Castellano, C., Cecconi, F., Loreto, V., & Parisi, D. (2004). Defining and identifying communities in networks. *Proceedings of the National Academy of Sciences of the United States of America*, *101*(9), 2658–2663.

Rand, W. M. (1971). Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, *66*(336), 846–850.

Roul, R. K., Asthana, S. R., Shah, M., & Parikh, D. (2016). Detecting spam web pages using content and link-based techniques. *Sadhana*, *41*(2), 193–202.

Scott, J. (2017). *Social network analysis*. Sage.

Seong, D. S., Kim, H. S., & Park, K. H. (1993). Incremental clustering of attributed graphs. *IEEE transactions on systems, man, and cybernetics*, *23*(5), 1399–1411.

Shafer, G. (1976). *A mathematical theory of evidence* (Vol. 1). Princeton university press Princeton.

Smets, P. (1990). The combination of evidence in the transferable belief model. *IEEE Transactions on pattern analysis and machine intelligence*, *12*(5), 447–458.

Smets, P. (1993). Belief functions: The disjunctive rule of combination and the generalized bayesian theorem. *International Journal of Approximate Reasoning*, *9*(1), 1–35.

Smets, P. (1995). The canonical decomposition of a weighted belief. In *Ijcai* (Vol. 95, pp. 1896–1901).

Smets, P. (2005). Decision making in the tbm: the necessity of the pignistic transformation. *International Journal of Approximate Reasoning*, *38*(2), 133–147.

Stringhini, G., Kruegel, C., & Vigna, G. (2010). Detecting spammers on social networks. In *Proceedings of the 26th annual computer security applications conference* (pp. 1–9).

Trabelsi, A., Elouedi, Z., & Lefevre, E. (2016). Handling uncertain attribute values in decision tree classifier using the belief function theory. In *International conference on artificial intelligence: Methodology, systems, and applications* (pp. 26–35).

Travers, J., & Milgram, S. (1967). The small world problem. *Phychology Today*, *1*(1), 61–67.

Vuokko, N., & Terzi, E. (2010). Reconstructing randomized social networks. In *Proceedings of the 2010 siam international conference on data mining* (pp. 49–59).

Wang, P., Xu, B., Wu, Y., & Zhou, X. (2015). Link prediction in social networks: the state-of-the-art. *Science China Information Sciences*, *58*(1), 1–38.

Washha, M. (2018). *Information Quality in Online Social Media and Big Data Collection: An Example of Twitter Spam Detection* (PhD Thesis). Université Paul Sabatier, Toulouse, France.

Washha, M., Qaroush, A., & Sedes, F. (2016). Leveraging time for spammers detection on twitter. In *Proceedings of the 8th international conference on management of digital ecosystems* (pp. 109–116).

Wasserman, S., & Faust, K. (1994). *Social network analysis: Methods and applications* (Vol. 8). Cambridge university press.

Watts, D. J., & Strogatz, S. H. (1998). Collective dynamics of small-worldnetworks. *nature*, *393*(6684), 440.

Wei, D., Deng, X., Zhang, X., Deng, Y., & Mahadevan, S. (2013). Identifying influential nodes in weighted networks based on evidence theory. *Physica A: Statistical Mechanics and its Applications*, *392*(10), 2564–2575.

Xie, J., Kelley, S., & Szymanski, B. K. (2013). Overlapping community detection in networks: The state-of-the-art and comparative study. *Acm computing surveys (csur)*, *45*(4), 43.

Yager, R. R. (1987). On the dempster-shafer framework and new combination rules. *Information sciences*, *41*(2), 93–137.

Yang, L., Cao, X., He, D., Wang, C., Wang, X., & Zhang, W. (2016). Modularity based community detection with deep learning. In *Ijcai* (pp. 2252–2258).

Yang, Z., Wilson, C., Wang, X., Gao, T., Zhao, B. Y., & Dai, Y. (2014). Uncovering social network sybils in the wild. *ACM Transactions on Knowledge Discovery from Data (TKDD)*, *8*(1), 2.

Zachary, W. W. (1977). An information flow model for conflict and fission in small groups. *Journal of anthropological research*, 452–473.

Zadeh, L. A. (1999). Fuzzy sets as a basis for a theory of possibility. *Fuzzy sets and systems*, *100*(1), 9–34.

Zheng, X., Zeng, Z., Chen, Z., Yu, Y., & Rong, C. (2015). Detecting spammers on social networks. *Neurocomputing*, *159*, 27–34.

Zhou, K., Martin, A., & Pan, Q. (2018). A belief combination rule for a large number of sources. *Journal of Advances in Information Fusion*, *13*(2).

Zhou, K., Martin, A., Pan, Q., & Liu, Z. (2018). Selp: Semi-supervised evidential label propagation algorithm for graph data clustering. *International Journal of Approximate Reasoning*, *92*, 139–154.

Zhou, K., Martin, A., Pan, Q., & Liu, Z.-G. (2016). Ecmdd: Evidential c-medoids clustering with multiple prototypes. *Pattern Recognition*, *60*, 239–257.

Zhou, Y., Cheng, H., & Yu, J. X. (2009). Graph clustering based on structural/attribute similarities. *Proceedings of the VLDB Endowment*, *2*(1),

718–729.